mc

1

Die Mikrocomputer-Zeitschrift

6 DM · 50 öS · 6,80 sfr. · Mai/Juni 1981

Basic mit Labels

Strichcode-Programme

Sortieren in Pascal und Basic

Daten-verwirrung

Bauelemente für die gesamte Elektronik. Valvo.

Mikrocomputer-Familie 8400.

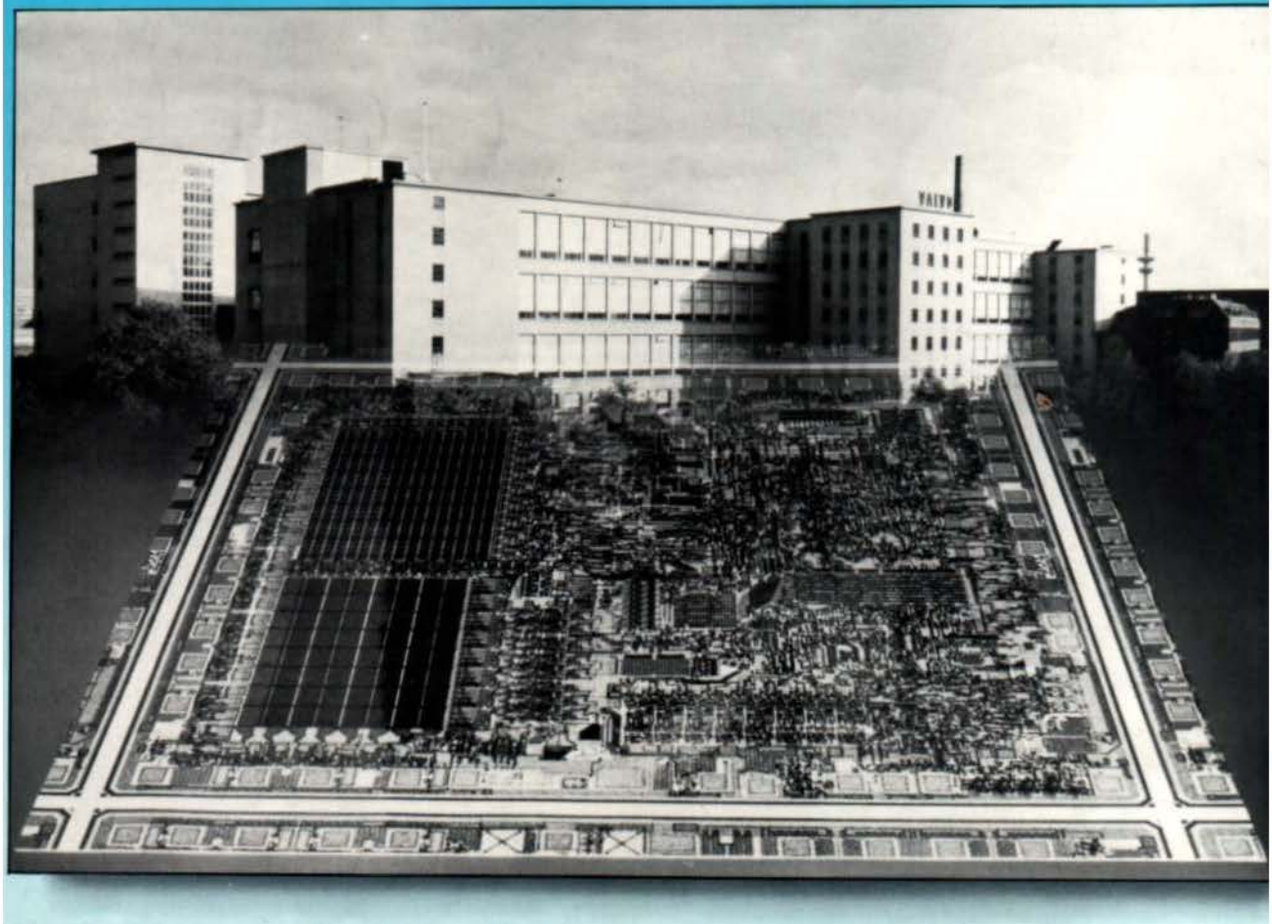
Basierend auf der bewährten Architektur des Mikrocomputers 8048, der bereits in Hamburg gefertigt wird, konzipiert und entwickelt Valvo die Mikrocomputer-Familie 8400, die für Steuerungsaufgaben u.a. in Haushalts-, Rundfunk- und Fernsehgeräten sowie für den Einsatz in Geräten der Industrie-Elektronik besonders angepaßt ist und sich durch ein gutes Preis/Leistungs-Verhältnis auszeichnet.

Folgende Versionen werden entwickelt:

	RAM (Byte)	ROM (KByte)
MAB 8400	128	Piggy Back
MAB 8405	32	0,5
MAB 8410	64	1
MAB 8420	64	2
MAB 8440	128	4

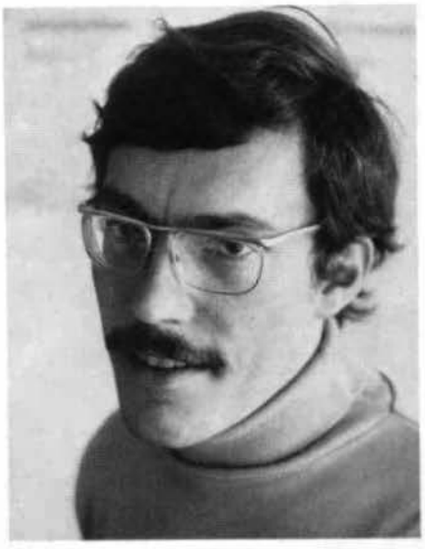
Zur Unterstützung bei der Programm-Entwicklung werden sowohl preiswerte als auch besonders leistungsfähige Entwicklungssysteme bereitgestellt.

Technische Beratung:
Produktbereich Mikroprozessoren
Burchardstraße 19, 2000 Hamburg 1
Telefon (040) 32 96–536



Bauelemente
für die gesamte
Elektronik

VALVO



Die mc-Historie

Das Thema, mit dem sich diese Zeitschrift beschäftigt, ist noch ziemlich jung: Erst vor zwölf Jahren, nämlich 1969, kam ein gewisser Ted Hoff im fernen Kalifornien auf die Idee, eine komplette Computer-Zentraleinheit auf einem Chip unterzubringen. 1970 wurde sein Plan mit dem „4004" Realität, einem einfachen 4-Bit-Mikroprozessor vergleichsweise geringer Leistungsfähigkeit.
Rund fünf Jahre später konnte man dann kompakte Computer mit einem Mikroprozessor als Zentraleinheit fertig kaufen; die „Personal-Computer"-Welle begann zu rollen. Dabei spielten übrigens Freizeit-Programmierer eine ebenso geburtsfördernde Rolle wie einst die Funkamateure in der Nachrichtentechnik.
Der Franzis-Verlag nahm sich der Materie schon früh an: Zahlreiche Grundlagenbeiträge erschienen in der ELEKTRONIK, die Serien „Praxis mit Mikroprozessoren" in der FUNKSCHAU und „Dem Mikroprozessor aufs Bit geschaut" in der ELO, ab Herbst 1977 auch eine ständige, achtseitige FUNKSCHAU-Rubrik „Mikrocomputer", ganz zu schweigen von acht Sonderheften und zahlreichen Büchern zu dieser Thematik.
Wegen des großen Erfolges dieser Publikationen hat sich der Verlag nun entschlossen, dem steigenden Informationsbedürfnis zum Thema „Mikroprozessoren" nachzukommen und eine spezielle Mikrocomputer-Zeitschrift herauszubringen. In diesem Jahr erscheinen insgesamt vier mc-Hefte im Abstand von jeweils zwei Monaten; 1982 lesen Sie mc dann monatlich. Sie haben das erste Heft in Händen und können selbst beurteilen, ob Ihnen mc zusagt; Kommentare und Anregungen sind der Redaktion stets willkommen.

Ihr Herwig Feichtinger

Gould Biomation K 100-D: „Weil µP-Systeme von morgen bereits zu schnell für andere Geräte sind!"

Und das im doppelten Sinn, denn immer komplexer werdende Hard- und Software-Probleme verlangen gerade in einer Zeit des explosionsartigen Fortschrittes nach Meßgeräten, die keine Probleme schaffen, sondern lösen.

Die zukunftsweisende Konzeption des K 100-D beinhaltet exklusive Eigenschaften wie 10 ns Auflösung, Messung über 16 Kanäle mit 1 K Speichertiefe und 100 MHz Abtastrate sowie die Möglichkeit des Datenvergleiches auch in der Timing-Darstellung.

Diese technische Überlegenheit ist gekoppelt mit einer extrem einfachen Bedienung. Wir machten die Erfahrung, daß Anwender, die nie zuvor mit einem K100-D gearbeitet hatten, diesen nach durchschnittlich 1–2 Stunden selbständig und ohne fremde Hilfe bedienen konnten.

Datenspeisekarte. Hier werden die Bedingungen vorgegeben, mit denen die Datenaufnahme erfolgen soll.

100 MHz Abtastrate

Der K 100-D ist ein Logik-Analyser der Spitzenklasse. Mit 100 MHz Abtastrate (!) ist er einer der schnellsten, was besonders Messungen von Laufzeitfehlern mit einer Auflösung von 10 ns (!) möglich macht.

16 Kanäle simultan

16 Kanäle (32 Kanäle optionell) stehen mit jeweils 1 K Speichertiefe bei 100 MHz Abtastrate zur Verfügung. Jeder Kanal ist individuell für TTL, ECL und variabel A+B (über das Keyboard) einstellbar, was Messungen an vier unterschiedlichen Logik-Familien gleichzeitig möglich macht.

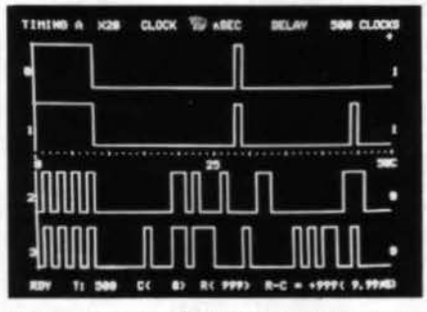
3 ns Latch Mode

Für die Fehlersuche nach sporadischen Störspitzen (Störspitzenerkennung 3 ns typisch) sowie nach Laufzeitfehlern steht ein Vergleichsspeicher mit den Betriebsarten A = B und A ≠ B zur Verfügung. Der Vergleichsspeicher arbeitet nicht nur, wie allgemein üblich, in Datendarstellung, sondern auch in der Timing-Darstellung, was besonders die Hardware-Analyse erheblich vereinfacht.

Ein umfangreiches Triggerpaket mit Takt- und Ereignisverzögerung sowie eine stufenweise Triggerung (Nested Trigger) sind selbstverständlich.

Self check

Um eine hohe Meßsicherheit zu garantieren, werden alle RAM-, ROM- und High Speed Memory-Funktionen beim Einschalten des Gerätes mit einem „Self-check" Programm überprüft.



Horizontal und vertikal gedehnter Ausschnitt eines 4-kanaligen Impulszuges.

Optionen

- General Purpose Interface Bus (IEEE 488)
- Erweiterung auf 32 Kanäle
- parallele Triggererweiterung auf 36 Kanäle
- serielle Triggererweiterung für RS 232 Schnittstellen
- µP und IEC Bus Adapter

Für weitere Informationen fordern Sie bitte das ausführliche Datenblatt an.

GOULD

Gould Instrument Systems
Dieselstraße 5-7, D 6453 Seligenstadt 3
Tel. 06182/21091-95, Tx. 4-184556
Niederlassungen in Österreich und der Schweiz

Berlin: 030/2115017-18
Bremen: 04298/2031, Tx. 2-45506
Stuttgart: 0711/702097-98
München: 089/364077-79, Tx. 5-28058

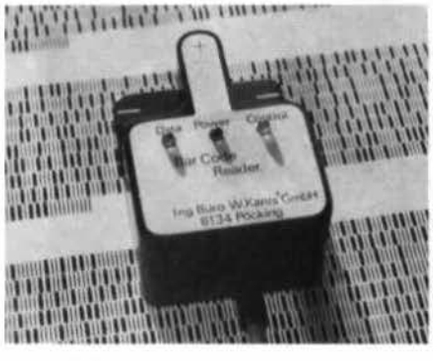
In diesem Heft

Datenverwirrung

Viele Normen schaffen auch Uneinheitlichkeit – bestes Beispiel hierfür sind die zahlreichen Codes zur Zeichendarstellung und Formate zur Datenweiterleitung. ASCII, Baudot, Hex, BNPF, B10F und ähnliches werden ausführlich erläutert ab **Seite 19**

Sortieren

Nicht selten dienen Tischcomputer dem Führen von Karteien, der Dateiverwaltung. Abhängig davon, um was für eine Art von Daten es sich handelt, sollte man sich genau überlegen, welche Algorithmen man für das Sortieren des Datenfeldes einsetzt. Mehr darüber auf **Seite 25**



Strichcode

Strichcodes, Balkencodes oder Bar Codes finden sich inzwischen auf vielen Verpackungen, können aber auch zur Verbreitung von Programmen dienen. Wie Strichcodes aufgebaut sind und wie das Einlesen in Computer per Software funktioniert, lesen Sie auf **Seite 33, 37, 42**

mc-kolumne

Die mc-Historie 3

mc-info 6

Impressum 81

mc-bücher 10

mc-grundlagen

Babylon's Datenverwirrung 19

Suchen und Sortieren in Pascal und Basic (1. Teil) 25

mc-hard

Der Strichcode 33

Der Volkscomputer kommt 45

Sharp MZ 80 B, eine Neuheit 55

mc-soft

Z-80-Programmiertip 24

Strichcode-Programm: Primfaktor-Zerlegung 37

Basic mit Labels 38

Apple II liest Strichcode 42

Der geknackte TRS-80 (1. Teil) 46

Große Sprünge mit dem 6502 51

ROM-Disassembler für PET und CBM 54

TRS-DOS mit Autostart 60

Pseudo-Reset beim CBM 3000 63

E-12-Reihe per Basic 66

mc-rechner

Lohn- und Einkommensteuer mit dem HP-67 56

TI-59-Besonderheiten 58

mc-applikation

Probleme des EDV-Unterrichts 62

mc-test

Mehr Komfort bei Apple-Disks 64

mc-markt 68

mc-vorschau 82

Siemens-Mikrocomputer-Kurse von Mai bis September

Die Münchner Schule für Mikrocomputer, Balanstr. 73, 8000 München 80, führt in den Monaten Mai bis September folgende Kurse durch:

Mai

4.–6.5.	ETA 85
4.–8.5.	Aufbaukurs 8086 Assembler 80/85
5.–8.5.	Grundkurs 85/80
11.–13.5.	ETA 80
11.–15.5.	Assembler 80/85 Assembler 86
18.–22.5.	PL/M 85/86 System-Software-Kurs Systemkurs 48
19.–22.5.	Grundkurs 48
25.–27.5.	ETA 85 ETA 86

Juni

1.–5.6.	Aufbaukurs 8086 Assembler 80/85 PL/M 85/86 System-Hardware-Kurs
2.–5.6.	Grundkurs 85/80
9.–11.6.	ETA 85
9.–12.6.	Grundkurs 85/80 Grundkurs 48
22.–26.6.	Assembler 80/85 Assembler 8086 PL/M 85/86
23.–26.6.	Grundkurs 85/80

Juli

29.6.–1.7.	ETA 85
29.6.–3.7.	Assembler 80/85 System-Software-Kurs
30.6.–3.7.	Grundkurs 48
6.–10.7.	Aufbaukurs 8086 Systemkurs 48
7.–10.7.	Grundkurs 85/80
13.–15.7.	ETA 85
13.–17.7.	Assembler 8086
14.–17.7.	Grundkurs 85/80
20.–22.7.	ETA 86
20.–24.7.	PL/M 85/86 System-Software-Kurs System-Hardware-Kurs
27.–31.7.	Aufbaukurs 8086 Assembler 80/85 Assembler 8086
27.–31.7.	Systemkurs 48
28.–31.7.	Grundkurs 85/80

September

31.8.–4.9.	Assembler 80/85
1.–4.9.	Grundkurs 85/80
7.–11.9.	Assembler 80/85 System-Software-Kurs
8.–11.9.	Grundkurs 48
14.–16.9.	ETA 86
14.–18.9.	Aufbaukurs 8086 Assembler 80/85 Systemkurs 48
15.–18.9.	Grundkurs 85/80
21.–23.9.	ETA 85
21.–25.9.	Assembler 80/85 PL/M 85/86
28.9.–2.10.	Assembler 8086 PL/M 85/86 System-Hardware-Kurs

Die Düsseldorfer Schule für Mikrocomputer (Neusserstr. 1, 4000 Düsseldorf) führt im gleichen Zeitraum folgende Kurse für Anfänger und Fortgeschrittene durch:

Mai

11.–15.5.	System-Software-Kurs System-Hardware-Kurs
18.–22.5.	Assembler 80/85
19.–22.5.	Grundkurs 48
25.–27.5.	ETA 85

Juni

1.–5.6.	PL/M 85/86
9.–12.6.	Grundkurs 85/80
22.–26.6.	Assembler 80/85 System-Software-Kurs

Juli

29.6.–3.7.	Aufbaukurs 8086
30.6.–3.7.	Grundkurs 85/80
6.–8.7.	ETA 85
6.–10.7.	PL/M 85/86
13.–17.7.	System-Software-Kurs
14.–17.7.	Grundkurs 85/80

August

24.–26.8.	ETA 85
24.–28.8.	Assembler 80/85

September

7.–11.9.	PL/M 85/86 System-Software-Kurs
15.–18.9.	Grundkurs 85/80
21.–23.9.	ETA 80
21.–25.9.	Assembler 80/85
28.–30.9.	ETA 85
28.9.–2.10.	System-Hardware-Kurs

3870-Anwenderbuch

Als Zusammenfassung bereits erschienener Datenblätter und Schriften über die Mikroprozessorfamilie F8/3870 brachte die Garchinger Firma Fairchild ein „µP-Applikationsbuch" heraus. Es enthält einen Überblick über vorhandene Entwicklungshilfen für diese Prozessoren, aber auch detaillierte Informationen über Einsatz und Bedienung des Einplatinen-Computers PEP 387. Schließlich folgen noch einige Beispielprogramme, so etwa zur Steuerung eines Matrixdruckers, zur seriellen asynchronen Datenübertragung, zur Realisierung einer Echtzeituhr, für die Ansteuerung eines LED-Displays, eines IEC-Buses und eines D/A-Wandlers sowie die Abfrage eines A/D-Wandlers. Die Programme wurden jeweils als gut kommentiertes Assembler-Listing abgedruckt.

Billig-Version des HP-85

Zugegeben, diese Überschrift ist etwas übertrieben: Gerade billig ist der HP-83 noch nicht gerade. Immerhin konnte der Preis gegenüber dem Tischcomputer HP-85 aber durch Weglassen des eingebauten Thermodruckers und des Kassettenlaufwerks deutlich gesenkt werden.
Der HP-83 ist damit für alle jene Anwender gedacht, die ohnehin mit einem externen Floppy-Disk-Laufwerk und einem externen Drucker arbeiten möchten – Peripheriegeräte, die auch von Hewlett-Pakkard selbst nun für HP-83 und HP-85 zu haben sind.
Eine Überraschung ist, daß HP nun plötzlich doch ein Assembler-ROM für diese beiden Tischcomputer anbietet und damit auf einen langgehegten Wunsch zahlreicher industrieller Anwender reagierte. Damit ist es z. B. möglich, schnelle Eingabe/Ausgabe-Routinen zu schreiben.

9900-Serie billiger

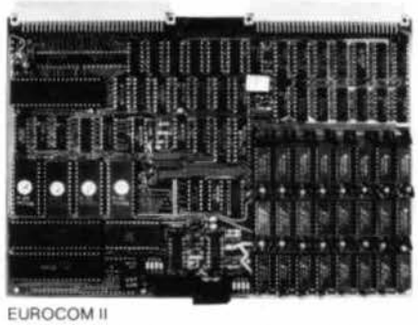
Texas Instruments hat die Preise für Bausteine der 9900-Prozessorfamilie drastisch um bis zu 50% gesenkt. Der Hersteller begründet dies mit Verbesserungen im fertigungstechnischen und technologischen Bereich. Der TMS 9900 gehört damit jetzt zu den preiswertesten 16-Bit-Prozessoren.
Am deutlichsten betrifft der Preisverfall die 9900-Peripheriebausteine, wie UART, Video-Controller, Bus-Controller und Floppy-Disk-Controller.

Software-Katalog

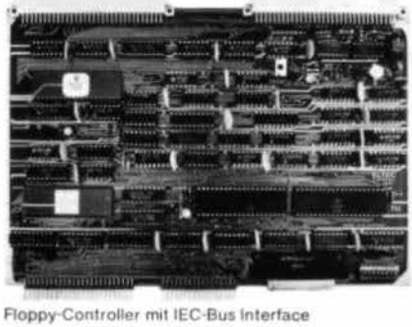
Einen herstellerunabhängigen Software-Katalog hat die Firma Profisoft (Robert-Koch-Str. 33, 7530 Pforzheim) herausgebracht. Er umfaßt Produkte von rund 50, meist professionellen Software-Herstellern für alle Arten von Mikrocomputern, z. T. auch für programmierbare Taschenrechner. Eingeschlossen sind Systemsoftware, kommerzielle Anwendungen, Branchen-Pakete und Spiele.
Für die gezielte Suche ist der Katalog nach unterschiedlichen Kriterien indiziert: Sachgebiet, Funktion, Hardware, Branche, Schlagwortverzeichnis. Allerdings ist die Software-Übersicht nicht ganz billig: 50 DM muß man für den Katalog hinblättern.

EUROCOM II

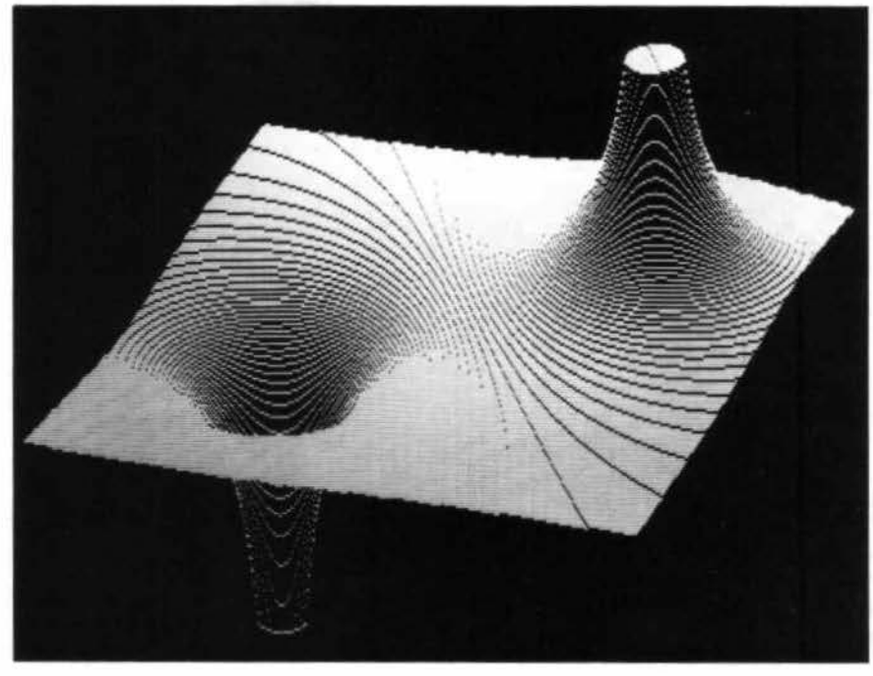
leistungsfähig, ausbaufähig, preiswert
auch EXORciser®-Bus kompatibel erhältlich



EUROCOM II



Floppy-Controller mit IEC-Bus Interface



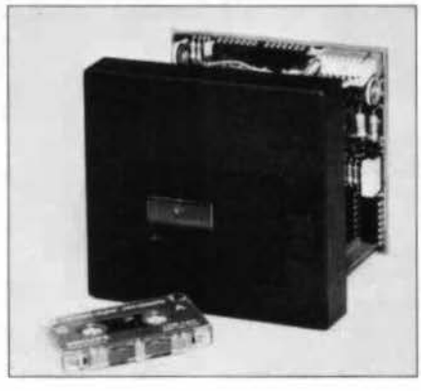
Der EUROCOM II
Vollgrafik 512 × 256 Bildpunkte CPU: MC 6809. 48 KByte RAM, 4 Steckplätze für EPROM. Schnittstellen: V 24, parallele I/O, Audio-Cassetten-Modem, Videoausgang. **DM 1.480,–** + MWST. (DM 1.672,40 incl. MWST.) 4 K Deburg-Monitor in EPROM, Bedienung über Standard-ASCII-Tastatur.

Weitere Ausbaumöglichkeiten:
EPROM-Programmierzusatz · Mit EXORciser-Bus erhältlich · Doppelte Auflösung 512 × 512 · RAM/Farberweiterung (96 KByte pro Zusatzkarte) · RAM-Erweiterung auf 240 KByte mit Paging möglich · Grafik bis zu 128 Farben/Graustufen · Fremdsynchronisation zur Einmischung des EUROCOM II - Videosignals in Videobilder · Low Cost Grafikdrucker. **Sonderanfertigungen:** EUROCOM II mit bis zu 1 MByte Bildspeicher zur Realtimeerarbeitung von Farbvideobildern.

Hochwertige kapazitive Tastatur

Software
für EUROCOM II mit Digital-Cassette.
Komfortabler Editor und Macro-Assembler DM 198,– + MWST. (DM 223,74 incl. MWST.)

Basic mit Grafikfunktionen, leistungsfähiges Basic 14 K, unterstützt voll die Grafik-Möglichkeiten des EUROCOM II. **198,–** + MWST. (DM 223,74 incl. MWST.).
Mehrseiten-Terminal-Programm, mit Softscroll über 16 Seiten u. Grafik-Unterstützung, z.B. Tektronics 4010 Emulation, **DM 310,–** + MWST. (DM 350,30 incl. MWST.).



Für EUROCOM II mit Floppy: Plattenbetriebssystem FLEX® BASIC, Extended BASIC, BASIC Precompiler, Editor, Macro Assembler, Wordprozessor. Neu: **PASCAL.** Kundenspezifische Software kann erstellt werden.

Floppy-Controller

Floppy-Controller-Platine zu EUROCOM II buskompatibel; für 5" und 8" Floppy-Drive; mit Floppy-Controller-Chip FD 1793; maximal 4 Drives, Double-Side; zusätzlich vollständiges normgerechtes IEC-Bus Interface.

DM 998,– + MWST. (DM 1.127,74 incl MWST.) Mit Direct-Memory-Access und Double Density **DM 1.498,– + MWST.** (DM 1.692,74 incl. MWST).

Mini-DCR
Preiswerter Massenspeicher.

Philips Mini-Digital-Cassetten-Recorder mit Interface zum direkten Anschluß an EUROCOM II.

80 KByte formatiert, 6000 Baud Aufzeichnungsrate. Treiber-Programm für 2 Drives auf EUROCOM II unterstützt Cassettenbetrieb. DM 490,- + MWST. (DM 533,70 incl. MWST.).

Kostenlosen Katalog anfordern.

Eltec Elektronik GmbH, Neubrunnenstr. 10
6500 Mainz, Postfach 1847,
Telefon 06131/23572, Telex 04187273

ELTEC

Holland
MANUDAX, Meerstaat 7,
5473 ZG Heeswijk (NB), Tel. 04139/2901
Belgien, Frankreich
MANUDAX, Rue Stephenson Straat 108-110
1020 Brüssel, Tel. 02/215.25.18

Schweiz
SPECTRALAB, Brunnenmoosstr. 7
8802 Kilchberg, Tel. 01/7155640
Österreich
POLYTRONIC, Wiener Straße 46, 4020 LINZ
Tel. 0732/60450

Schweden
ELEKTRIX, Bergsgaten 35, S-214 22 MALMÖ
Tel. 040 - 973736

Commodore wächst

Commodore International Ltd., mit seinen Tischcomputern Marktführer in der Bundesrepublik, setzt seine Expansion fort.
So gründete das in Produktion und Vertieb weltweit operierende Unternehmen, dessen Zentrale kürzlich von Kalifornien nach Norristown im US-Bundesstaat Pennsylvania verlegt wurde, Tochtergesellschaften in Norwegen und Dänemark sowie in Australien.
Im deutschen Commodore-Werk in Braunschweig ist zum Jahreswechsel planmäßig die Produktion von Tischcomputern für die Bundesrepublik und den europäischen Markt angelaufen.
Jack Tramiel, der bisher das Unternehmen als President leitete, ging in den Aufsichtsrat und ist dort Vicepresident und Chief-Executive Officer, wie AR-Vorsitzender Irving Gould jetzt in New York mitteilte. Tramiel hatte die Gesellschaft 1958 als Marketingorganisation für Schreibmaschinen und elektromechanische Büromaschinen gegründet.
Neuer Vorstandsvorsitzender wurde James Finke. Er war vor dem Wechsel zu Commodore Vicepresident bei Data General und als General Managar für die Aktivitäten dieses Computerherstellers in Europa vorständig. Auch in seinen früheren Positionen bei Motorola und der Medical Division von General Electric war er für das internationale Geschäft verantwortlich.

HP-41 mit fünffacher Kapazität

Unter der Bezeichnung HP-41 CV stellte Hewlett-Packard eine neue Version des HP-41 vor, die bis zu 2000 Programmzeilen speichern kann. Damit ist es nicht mehr erforderlich, die Kapazität durch Einstecken von externen Speichermoduln zu erweitern, so daß die Steckplätze für Peripheriegeräte frei bleiben. Alternativ läßt sich der Speicher auch in Form von 300 Datenregistern verwenden.
Möglich wurde der Fortschritt

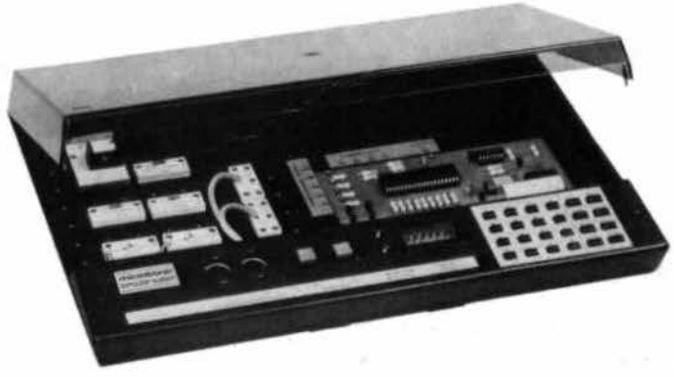
durch Verwendung von höherintegrierten CMOS-Speicherbausteinen, die nun auch in den „alten" HP-41 C eingebaut werden, wodurch eine Preissenkung des HP-41 C möglich wurde.

4-Bit-Lerncomputer

Obwohl 4-Bit-Mikroprozessoren heute nur noch wenig eingesetzt werden, eignen sie sich hervorragend, um Neulingen die Computertechnik näherzubringen: Der Befehlssatz ist gut überschaubar, der Hardware-Aufwand ist ziemlich gering, und der Computer arbeitet immer ziffernweise parallel, d. h. die Datenbreite entspricht genau einer Dezimal- oder Hexadezimal-Ziffer.
Ein Lerncomputer, der im Herbst dieses Jahres lieferbar sein wird und mit dem 4-Bit-Prozessor TMS 1600 arbeitet, ist das MICROtronic-System 2090 (*Bild*). Das IC TMS 1600 enthält die Zentraleinheit, einen 512 Byte großen Arbeitsspeicher und ein 4-KByte-Monitorprogramm. Weitere 512 Byte RAM-Programmspeicher sind in einem IC 2114 untergebracht.
Besonderer Wert wurde auf ein didaktisch günstig aufgebautes Begleitmaterial gelegt. Das MICROtronic-System 2090 wird ab Herbst über den Franzis-Verlag beziehbar sein.

TI-99/4 jetzt mit PAL-Modulator

Texas Instruments stellt den Heimcomputer TI-99/4 jetzt nicht mehr nur in der amerikanischen Farbfernsehnorm NTSC, sondern auch in der deutschen PAL-Norm mit UHF-Modulator (FTZ-geprüft) her. Dadurch ist es nicht mehr erforderlich, einen besonderen NTSC- oder Mehrnormen-Farbfernsehempfänger zur Wiedergabe zu verwenden. Das abgegebene Signal liegt im Bereich VHF/Band 1, Kanal 4. Die Empfangsantenne kann „durchgeschleift" werden, so daß kein Umstecken zwischen Computer und normalem Fernsehempfang erforderlich ist.
Für alle, die sich ihre Programme gern selbst schreiben, kam ein ROM-Modul „Extended Basic" heraus, und in Verbindung damit kann auch ein zusätzlicher Arbeitsspeicher angeschlossen werden, der das verfügbare RAM von 16 KByte auf 48 KByte vergrößert. Die Produktpalette wird noch mit Einsteckmoduln für Text- und Dateiverwaltung, Lagerverwaltung, Rechnungstellung und Terminal-Simulation erweitert; in Kürze soll auch ein „universelles Karteisystem" folgen.



Doch nur 4 Computer pro Schule

Entgegen der ursprünglichen Vorstellung, jedes mathematisch-naturwissenschaftliche Gymnasium und jede Fachoberschule des technischen Zweiges mit acht Tischcomputern für den Informatik-Unterricht auszustatten, mußte das Bundesland Bayern seine in dieser Richtung extrem progressive Politik etwas revidieren. Die Vorstellungen des Staatsinstituts für Schulpädagogik in München laufen im Augenblick darauf hinaus, jeder interessierten Schule (also nicht unbedingt allen) drei Computer vom Typ CBM 3008 und einen CBM 8032 zu genehmigen.
Die Auswahl von Commodore-Computern ergab sich nach einer öffentlichen Ausschreibung und wird auf den verhältnismäßig günstigen Preis zurückgeführt.
An bayerischen Schulen wird der Computer derzeit vorwiegend im Informatik-Wahlunterricht in den 10. Klassen und in der Kollegstufe sowie auch in freiwilligen Schüler-Arbeitsgemeinschaften genutzt, z. T. schon in der 6. und 7. Klasse. Die Akademie für Lehrerfortbildung führt regelmäßig Kurse für interessierte Lehrer durch, ebenfalls an Computern des CBM-Typs.

Sinclair ZX80 Microcomputer.

Der programmierte Erfolg für Studium, Beruf und Freizeit.

NEU! FÜR NUR 498.– DM

Computer prägen in zunehmendem Maße nahezu alle Bereiche des täglichen Lebens. Und wer in Studium oder Beruf Erfolg haben will, muß sich mit ihrer Sprache und Bedienung vertraut machen.

Sinclair ZX80. Der einfachste Weg zur Computertechnik. Der ZX80 Microcomputer verarbeitet „Basic", die am weitesten verbreitete Computersprache. Seine Speicherkapazität bietet schon in der Standardausführung 1 k-Byte RAM und 4 k-Byte ROM. Kombiniert mit einem einzigartigen Lernprogramm, führt Sie der ZX80 Schritt für Schritt in das Gebiet der Datenverarbeitung ein. Systematisch. Gründlich. Unkompliziert.

Das ZX80 Basic-Handbuch (deutsch). Die Grundlage Ihres programmierten Erfolges. Jedem theoretischen Kapitel folgt stets eine praktische Lektion. Von der Einführung bis zu schwierigen Programmen.

System ZX80. Leicht zu bedienen. Den kompakten ZX80 Microcomputer (nur 174 mm x 218 mm) einfach an Ihren handelsüblichen Fernseher (UHF, Kanal 36) anschließen und mit einem normalen Cassettenrecorder verbinden. Fertig ist Ihr persönliches Computer-Terminal! Sämtliche Verbindungskabel inkl. Netzgerät werden mitgeliefert. Das Arbeiten mit dem ZX80 ist denkbar unkompliziert. Z. B. durch seine übersichtliche Tipptastatur (mit 22 graphischen Symbolen). Sinnvolle Basic-Abkürzungen. Und besonders leistungsfähige, neue „LSI-Microchips" bzw. „Super-ROM"!

Das 16 k-Byte RAM-Erweiterungsmodul ist rückseitig an der Steckerleiste des ZX80 leicht anzubringen.

Ausbaufähig für Könner. Die Speicherkapazität läßt sich mit dem „RAM-Modul" schnell auf 16 k-Bytes extern erweitern, die Kapazität im Basic-Bereich intern durch Austausch von „ROM-Modul" und Tastenfeld auf 8 k-Bytes.

Für Experten: „Peek" und „Poke" ermöglichen die Eingabe von Maschinencode-Instruktionen. Der „USR"-Befehl bewirkt den Sprung zu einem in Maschinensprache geschriebenen Unterprogramm. Das Gerät bewältigt bis zu 26 „Strings" jeder Länge, die sich alle in Beziehung zueinander setzen lassen. Negative Darstellung aller Zeichen ist möglich. Und der einmalige „Syntax Check" sorgt für absolut fehlerfreie Programmeingabe. Kein Wunder also, daß führende Fachzeitschriften bestätigen: Der Sinclair ZX80 bietet ein sehr gutes Preis-/Leistungsverhältnis! Wir wünschen viel Spaß beim Programmieren!

Sinclair Research Ltd. Deutschland, Erlenweg 2, Postfach 1710, 8028 Taufkirchen b. München, Tel. (089) 6 12 17 93

sinclair ZX80

Coupon Sinclair Research Ltd. Deutschland
Erlenweg 2, Postfach 1710, 8028 Taufkirchen b. München mc 1/81

Bitte senden Sie mir _____ Exemplar(e) ZX80 Microcomputer (à DM 498,–) inkl. Zubehör

und _____ Exemplar(e) 16 k-Byte RAM-Erweiterungsmodul Speicher (à DM 249,–)

und _____ Exemplar(e) 8 k-Byte ROM-Erweiterungsmodul (à DM 98,–)

Preise inkl. MwSt., Porto und Verpackung (6 Monate Garantie)

Summe insgesamt DM _____ Versand per Nachnahme oder Scheck im voraus.

Name _____

Straße _____ PLZ, Ort _____

Unterschrift _____ Datum _____

Pascal

Einführung für Techniker. Von Wolfgang Schneider. 146 Seiten. Lwstr.-geb. 22 DM. Vieweg-Verlag, Wiesbaden.
ISBN 3-528-04181-1

Die Programmiersprache Pascal gewinnt in letzter Zeit auch im Mikrocomputer-Bereich an Bedeutung. Sie ist zwar, und das werden auch Pascal-Fans zugeben, deutlich schwieriger zu erlernen als z. B. Basic, zwingt jedoch auch bei längeren Programmen zu einer übersichtlichen Strukturierung. Für mehrere Tischcomputer sind heute bereits Pascal-Interpreter oder -Compiler lieferbar.
„Pascal" von Wolfgang Schneider setzt praktisch keine Vorkenntnisse voraus. Der Autor beschreibt daher zunächst die allgemeinen Grundlagen der Datenverarbeitung, vergleicht kurz unterschiedliche Programmiersprachen und widmet sich endlich der Problemaufbereitung für das Programmieren in Pascal. Sprachelemente, Vereinbarungen, Zuweisungen, Ein- und Ausgabe sowie Steueranweisungen werden anhand zahlreicher Beispiele besprochen. Diese Beispiele sind zwar programmtechnisch in keiner Weise optimiert, fördern das Verständnis aber wesentlich mehr als graue Theorie allein.
Unter Berücksichtigung der sich bei Pascal fast automatisch ergebenden Strukturierung enthält das Buch Programmablaufpläne übrigens nicht in Form von Flußdiagrammen, sondern als „Struktogramme". Fe.

Datenverarbeitung mit Mikroprozessoren

Teil 2: Software. Von Bodo Richard. 176 Seiten, ferner 12 Seiten auf Falttafeln, kart. 28 DM. Carl-Hanser-Verlag, München/Wien.
ISBN 3-446-12789-5

Der Autor, selbst Oberstudienrat, hat hier ein Lernbuch im wahrsten Sinn des Wortes erstellt: Eine große Zahl von Beispielen, Übungen und lernzielorientierten Tests sind typisch dafür. Unvermeidlicherweise muß sich ein solches Buch mit einer ganz bestimmten Prozessorfamilie beschäftigen – hier mit dem 8080/8085. Dies macht natürlich die Programmierbeispiele für die Benutzer anderer Computer weitgehend unverständlich; bestenfalls Z-80-Fans können noch Nutzen daraus ziehen, ein Problem, das sich so leicht nicht ändern läßt. Das Buch beginnt mit einem Kapitel über die binäre Zahlendarstellung in Computern, erläutert dann die Ausführung der Grundrechenarten im Binärsystem, wendet sich der realen Programmierung dieser Dinge zu und bespricht schließlich allgemeine Grundsätze der Programmerstellung. -ger

Mikrorechner

Wirkungsweise, Programmierung, Applikation. Von Dr. Wolfgang Schwarz, Dr. Gernot Meyer und Dr. Dietrich Eckhardt. 360 Seiten, Kunststoffeinband, DDR-Preis 32 M. VEB Verlag Technik, Berlin.

Der Begriff „Mikrorechner" läßt gleich erkennen, aus welchem Teil Deutschlands das vorliegende Buch kommt; er steht für das gleichbedeutende Wort „Mikrocomputer".
In der DDR sind inzwischen mehrere westliche Mikrocomputer-Chips mit Erfolg nachgebaut worden, so z. B. der Prozessor 8080, auf dessen Befehlssatz sich die Autoren bei ihren Programmierbeispielen auch beziehen und der die DDR-Bezeichnung K580IK80 trägt.
Wie schon beim Wort „Mikrorechner" wurde versucht, erläuternde Texte möglichst von englisch-deutschem Kauderwelsch freizuhalten – eine Bemühung, die sicher auch mancher hierzulande erscheinenden Publikation nicht schaden würde. Was das Buch von manch anderem abhebt, ist die Gründlichkeit, mit der die Autoren die einzelnen Befehle der CPU, typische Programmiertechniken und arithmetische Algorithmen besprechen. Man fragt sich, warum 8080-Hersteller Intel nicht selbst in der Lage war, eine so exakte und gründliche Befehlsbeschreibung zu liefern... Fe.

Basic für Mikrocomputer

Geräte – Begriffe – Befehle – Programme. Von Herwig Feichtinger. 256 Seiten, zahlreiche Bilder, Tabellen und Programme. Kart. 26 DM. Franzis-Verlag, München.
ISBN 3-7723-6821-2

Dieses Buch mit genau 2^8 Seiten zeigt sich in mehrfacher Hinsicht von gegensätzlichen Seiten: Es ist aktuell und zeitlos, produktspezifisch und doch allgemein, es informiert gleichermaßen den Anfänger wie den Fortgeschrittenen. Voraussetzung dafür ist einerseits die klare Gliederung, andererseits die erfreulich knappe Sprache, die gesammeltes Wissen in konzentrierter Form vermittelt. Im aktuellen Teil beschreibt der Autor die wichtigsten Basic-Computer, die für private Zwecke in Frage kommen. Der Einsteiger kann hier die Eigenschaften verschiedener Modelle vergleichen und seine Kaufentscheidung davon abhängig machen. Dem Fortgeschrittenen nützt dieses Kapitel beispielsweise dann, wenn er ein Basic-Programm an seinen eigenen Computer anpassen will, da er zu jedem besprochenen Modell Angaben über Befehlsvorrat, Speicheraufteilung usw. findet. Ein umfangreicher Nachschlageteil erweitert das Buch zum Taschenlexikon, das mit praktischen Hinweisen gespickt ist – vorzugsweise für 6502-Maschinen. Die folgenden Kapitel befassen sich schließlich mit der Sprache Basic. Wie in den vorangegangenen Abschnitten verfährt der Autor auch hier nach dem Rezept: keine lehrbuchhafte Einführung, sondern kurze, prägnante Information. Ein Buch also für Wissensdurstige mit wenig Zeit.
Ho.

4-Bit Mikrocomputer mit Familienzuwachs.

Zwei neue 4 Bit CMOS-Mikrocomputer erweitern die komfortable 4 Bit-Familie 7500 von NEC. Ausgestattet mit integrierten LCD-Treibern, einem Befehlsatz von bis zu 92 Instruktionen, serieller I/O (z.B. für A/D Konverter), bis zu 32 I/O-Linien, eingebautem Zeitgeber und Zähler, mit wählbaren RAM's und ROM's; natürlich in CMOS mit Standby-Modes und für Spannungen von 2,7 V bis 5,5 V ist die 4-Bit µC-Familie, µPD 7500 aufwärtskompatibel, erweiterungs- und anpassungsfähig. Sie zeichnet sich durch ihre Vielseitigkeit und geringe Leistungsaufnahme aus (CMOS und Standby).

4-Bit Mikrocomputer mit variablen Speichergrößen zeichnen sich durch ein extrem günstiges Kosten-/Leistungsverhältnis aus und sind daher ideal für den Geräte-Entwickler und -Hersteller. Sie sind ideal für alle Großserienentwicklungen in der Konsumelektronik – Foto, Film, Radio, Kassette, Video – ebenso wie in der elektronischen Nachrichten- und Meßtechnik, z.B. bei Telefonanzeigen, Citizenbandgeräten, LCD-Multimetern oder medizinischen Geräten und Personenrufanlagen.

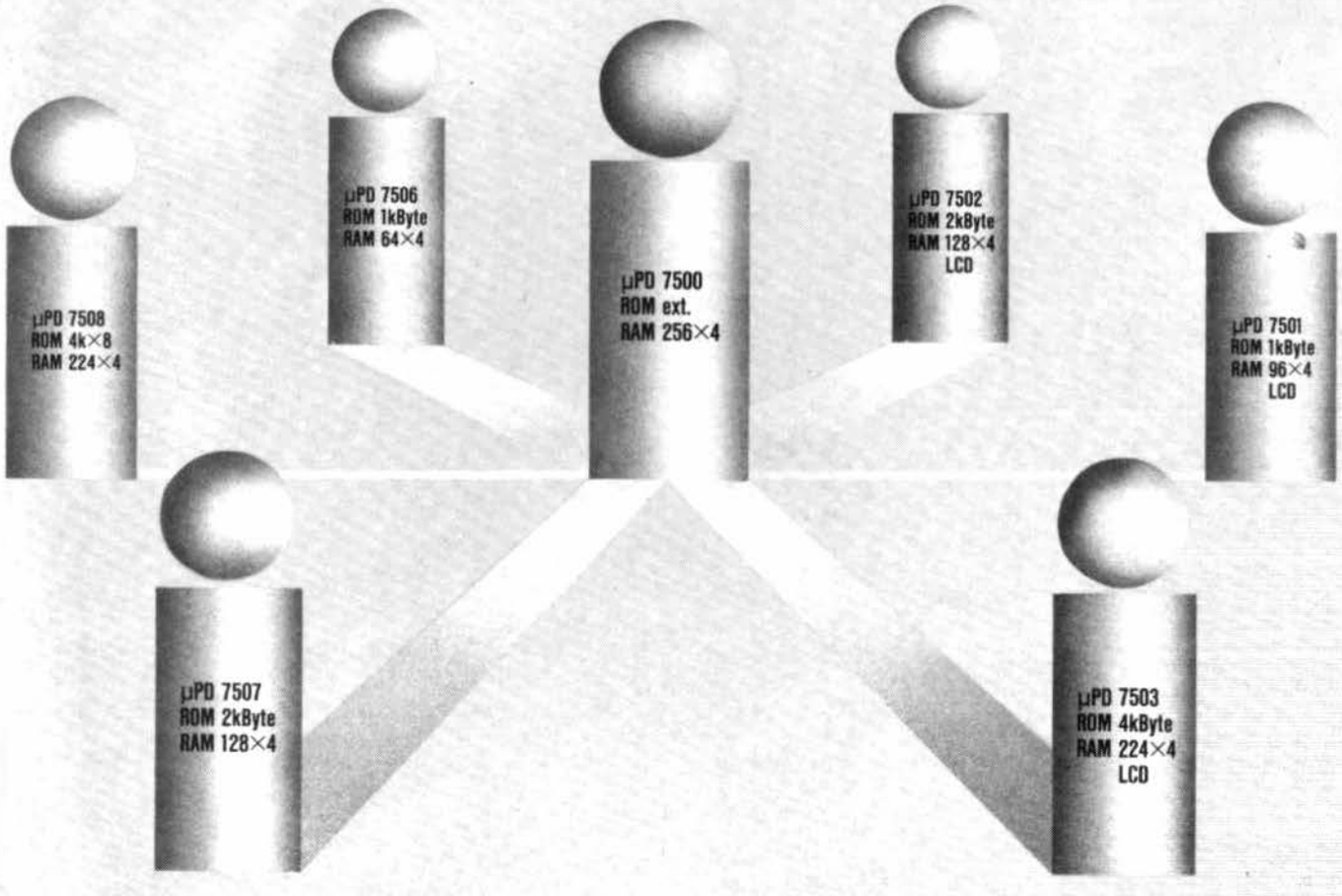
Der Komfort und die Qualität unserer µPD 7500er-Familie wird Sie überzeugen und bietet Ihnen genügend Freiraum für zukünftige Entwicklungen.

NEC Electronics (Europe) GmbH
Karlstraße 123–127, 4000 Düsseldorf
Tel. (0211) 36141, Tx. 8587419 nec d

Niederlassungen:
NEC Electronics (Europe) GmbH
Solitudestraße 218, 7000 Stuttgart 31
Tel. (0711) 881119, Tx. 7252220 nec d

NEC Electronics (Europe) GmbH
Gutenbergstraße 4, 3012 Langenhagen
Tel. (0511) 738001, Tx: 9230109 nec d

NEC Electronics (Europe) GmbH
Bayerstraße 21, 8000 München 2
Tel. (089) 591364, Tx. 522971 nec d



NEC – Die bessere Technologie.

Mikroelektronik

Eine Technik macht Epoche. Von Kurt Garbrecht und Alfred Prommer. 76 Seiten, kart., kostenlos unter Best.-Nr. L 43/1314 zu beziehen von Siemens, ZVW 85, Gründlacher Straße 258, 8510 Fürth-Bislohe.

Das kleine Bändchen eignet sich auch für Leute ohne Vorkenntnisse auf dem Gebiet der Elektronik und behandelt vor allem technologische Aspekte der Mikroelektronik, aber auch Fragen der Einsatzmöglichkeiten. Unter anderem fanden wir zwei recht interessante Aussagen darin: Trotz der heutigen Verbreitung elektronischer Speicherbausteine entspricht die Gesamtzahl der integrierten Transistorfunktionen auf der ganzen Welt erst etwa der Kapazität eines einzigen menschlichen Gehirns. Und: Wozu brauchen wir Größtintegration mit 10 Millionen Transistoren auf einem Chip? Nun, so die Autoren, das ist keinesfalls unsinnig, denn erst damit wird es wirtschaftlich möglich sein, wirklich technisch-intelligente Systeme durch Redundanz funktionssicher zu machen. -ger

8080A/8085 Programmieren in Assembler

Von Lance E. Leventhal. 463 Seiten mit zahlreichen Tabellen und Abbildungen. Kart. 49 DM (deutsch). Tewi-Verlag, München.
ISBN 3-921803-03-9

Der Mikroprozessor 8080 war praktisch der erste weiter verbreitete, frei erhältliche Prozessor und ist auch heute noch trotz vielfältiger Konkurrenz so etwas wie ein Industriestandard. In zahlreichen Lehr- und Lerncomputern dient er dazu, Neulingen erstes Wissen um Software und Hardware zu vermitteln.
Und trotz aller höheren Programmiersprachen ist die Assembler-Programmierung nach wie vor durch nichts zu ersetzen. Nur, bisher gibt es relativ wenig gute Literatur über Assembler-Programmierung in deutscher Sprache. Eine solche Lücke füllt die nun vorliegende deutsche Übersetzung des Buches von Lance E. Leventhal. Wer ein gewisses Grundwissen über die Mikrocomputertechnik mitbringt und mehr mit seinem 8080-System anfangen möchte, als nur stur Bytes abzutippen, ist mit diesem Buch gut beraten.
Leventhal setzt beim Leser das Vorhandensein eines symbolischen Assemblers voraus: Abgedruckt werden bei Programmbeispielen nur die mnemonischen Befehlskürzel, nicht aber die (hexadezimalen) Objektcodes. Dies ist aber durchaus sinnvoll, um nicht zu systemabhängig zu sein. -ger

16-Bit-Generation Z 8000

Aufbau und Anwendung. Von Peter Stuhlmüller. 1. Auflage, 464 Seiten, Hardcover mit Schutzumschlag, 73 DM. Te-Wi-Verlag, München.
ISBN 3-921803-07-1

Obwohl nach einhelliger Meinung der Experten 16-Bit-Mikroprozessoren im Vergleich zu ihren 8-Bit-Kollegen nur einen vergleichsweise geringen Marktanteil einnehmen werden, nimmt ihre Bedeutung rasch zu. Die beiden derzeit hochentwickeltsten Typen dürften 68000 (Motorola) und Z 8000 (Zilog) sein, und letzterer ist Hauptgegenstand des Buches von Peter Stuhlmüller. Es beschäftigt sich ausgiebig mit der Hardware-Struktur eines Computers mit dem Z8000, beschreibt die zur Verfügung stehenden Befehle und Adressierungsarten und zeigt auch die besonderen Speicherzugriffs-Möglichkeiten des Z 8010. Lobenswert ist, daß Stuhlmüller auch einen Blick zur Seite wagt und seinen „Favoriten" mit anderen Konkurrenten, insbesondere dem 68000, objektiv vergleicht. Das Buch schließt mit der Beschreibung eines Multiprozessor-Systems, eines Parallelrechners also, mit Z-8000-Bausteinen. Fe.

The Basic Handbook

Von David A. Lien (engl.). 360 Seiten. Kart. 44 DM. Verlag Interface Age GmbH, Vaterstetten b. München.
ISBN 0-932760-00-7

David A. Lien vergleicht einen Basic-Computer mit dem Volkswagen: Noch vor wenigen Jahren sahen elitäre Gruppen auf ihn als etwas Minderwertiges herab – und heute gibt es mehr Computer für Basic als für irgendeine andere Programmiersprache. Bedauerlicherweise versteht aber jeder Computerhersteller unter Basic etwas anderes, obwohl es auch hierfür eine Norm gibt: den ANSI-Standard. Jedenfalls versuchte Lien, die Syntax und Wirkungsweise von Befehlen unterschiedlicher Dialekte lexikonartig zu vergleichen.
Für jeden Befehl findet sich deshalb ein „Test Program", ein „Sample Run", Angaben über „Variations in Usage" und Querverweise für vergleichbare Befehle. Sehr nützlich ist die manchmal auftauchende Bemerkung „If your computer doesn't have it...", die es erlaubt, auf dem eigenen Computer nicht vorhandene Befehle irgendwie zu umschreiben.
Was das Buch nicht ist und nicht sein will: Ein Basic-Lehrbuch, ein Ersatz für ein gutes Computer-Handbuch des Herstellers und ein qualitativer Vergleich von Basic-Computern. Fe.

HOFACKER

Software für PET/CBM

Spiele für PET und CBM
*nur auf 8K PET alite ROMs

Best.-Nr.	Titel	Preis/DM
401	U-Bootjagd für PET	49,00
402	Einarmiger Bandit	29,00
403	Black Jach (17 und 4)	49,00
404	Joystick mit Programm *	149,00
405	Biorythmus für PET	49,00
406	Krieg der Sterne	29,00
407	Krieg der Sterne mit Abschießen	29,00
408	Las Vegas für PET	29,00
420	Schach für PET und CBM	79,00
421	Linear Joystick *	199,00
427	Reversi für PET	29,80
428	Funktionsgenerator *	19,80
4281	Platinenbausatz dazu	79,00
429	GAMEPAC I	29,80
430	GAMEPAC II	29,80
432	Stimulating Simulations	49,00
433	Astrologieprogramm	39,80
434	Bridge-Programm für PET	49,00
443	Dual Joystick, ATARI	279,00
447	Computerspiele	19,80
448	Casino und Spielsimul.	19,80
460	57 Programme (3 Cassetten)	87,00
414	Partyprogramm	49,00
453	Diät- und Gesundheitsprogramm	19,80
472	Seawolf	29,80
478	Autorennen *	49,00
4813	Joystick für PET und CBM	79,00
443	Dual Joystick PET/CBM	279,00

Geschäftsprogramme für PET

426	Text Processing Syst.	96,00
437	General Ledger-Hustler	69,00
438	Checking Account (dt.)	79,00
439	Trust Account	69,00
440	Legal Diary	69,00
441	RENT Accounts	69,00
446	Druckerinterface für RS 232 (426 Software dazu)	669,00
476	Commerzielle Programme	89,00
477	Mathematische Programme	49,00
4761	Adreßverwaltung CBM	299,00
4781	Fakturierung CBM	399,00
479	Lagerverwaltung CBM	249,00
498	Banküberweisungen	49,00
499	Inventur PET *	49,00
4825	Bidirekt. RS 232 Schnittstelle	899,00

Assembler, Monitore, Utilities, Programme in Maschinensprache

461	JANA Monitor *	29,80
462	Joystickprogramm	29,80
473	JANA für 16K/alte ROMs	49,00
474	JANA für 32K/alte ROMs	49,00
475	Assembler für 16/32K	98,00
431	Progr.-Experimente PET	39,80
417	Hex-Monitor für PET *	19,80
418	Super-Monitor für PET *	19,80
425	Assembler für 8K	98,00
449	Programmieren in Maschinenspr.	39,80
450	Ein-/Ausgabe-Programmierung	49,80
454	220V/50 Hz Schalterinterface	169,00
455	Externe Experimentierplatine	199,00
456	Analog-Digital / Digital Analog-Wandler, Software	499,00
492	Ultra Monitor CBM	29,80
493	Ultra Monitor PET *	19,80
494	Renumber CBM	29,80
495	BASIC Linker, PET *	29,80
496	Renumber PET *	19,80
497	TRACE + 3 Utilities *	19,80
4812	Editor/Assembler in Masch.-Spr.	169,00

Lehr- + Lernprogramme, Haushaltsfinanz., Computermusik

416	Musikprogramm PET	49,00
419	Musikplatine + Software, Baus.	178,00
435	Computermusik PET	39,80
444	Musik-Box	199,00
457	Audio-Cassette mit Musik PET	9,80
458	Computer-Musik (Audio)	19,80
459	Computer-Musik (Schallplatte)	19,80
415	Lernprogramm (Educator 1)	29,80
4151	Lernprogramme (Educator 2)	39,00
422	Programmier-Tricks PET	39,00
423	BASIC-Kurs für PET	99,00
424	Graphik + Bewegung für PET	29,00
436	Morse-Trainer	29,80
451	Advanced BASIC	69,00
409	Haushalts-Utilityprogramm 1	29,00
410	Haushalts-Utilityprogramm 2	29,00
411	Finanzprogramm 1 + 2	138,00
413	Finanzprogramm 3	69,00

Neueste Programm-Cassetten für PET/CBM

478 + 433	Autorennen incl. Dualjoyst.	328,00
489	Adventureland	29,80
4814	Pirate Adventure (24K)	69,00
4815	Reisekostenabrechnung CBM	199,00
4816	Mathematikpaket ca. 30 Progr.	199,00
4817	Memdata (Adressl.) PET	49,00
4818	Small Business Progr.	19,80
4819	Engl. Grammatik PET	29,80
4820	Fakturierprogramm	199,00
8221	Utilities für PET (8K)	19,80

Leercassetten – C 10 –

8089	1 Cassette	3,50
8100	10 Cassetten	29,80
8096	100 Cassetten	249,00

Best.-Nr.	Titel	Preis/DM
Apple		
6006	DATA-Management	78,00
6017	Inventurprogramm	299,00
6014	The Basic Teacher	84,00
6011	Invoicing	189,00
6002	Priv. Sekretärin	189,00
6015	Billing Management	299,00
6016	Retail Management	189,00
6010	Asset Record	189,00
6007	Progr. Gymnastik	63,00
6111	Mailing-List	249,00
6001	Prof. Sekretärin	299,00
6013	Word Processor	299,00
6110	Apple Sargon (C)	110,00
6118	Apple Sargon (D)	119,00
6119	Super FORTH	169,00

Neue Apple Software D=Diskette, C=Cassette

6120	Reversal (D)	129,00
6126	Dateiverwaltung (D)	199,00
6127	Adressenverwaltung (D)	199,00
6128	Super Invaders (D)	49,00
6129	PASCAL Programme (D)	29,80
6230	Utilities I (D)	99,00
6131	Utilities II (D)	99,00
6132	Statistik (D)	99,00
6133	Inventory (D)	69,00
6134	Invoicing (D)	79,00
6135	Dictionary (D)	49,00
6136	Game Package (D)	69,00

NEUHEITEN

600	Diskettenhüllen	2,30
601	Plastikordner Redysoft	19,80
602	ELCOMP-Plastikordner	19,80
603	ELCOMP-Sammelordner	14,80
604	Ordner mit 20 Diskettenhüllen für 40 Disketten	69,00
605	DIN A5 ELCOMP-Plastikordner	9,80

Platinenservice und Software für Ohio Scientific

8234	EPROM BURNER f. Superboard (Platine m. Softw. u. Anleitung)	149,00
8235	6522 Karte für C1P od. Superb.	89,00
8236	EPROM-Karte für C1P + Superb.	59,00

Bauanleitungen für Superboard

0300	Bauanl. f. Druckeranschl. (RS 232)	5,00
0301	Grafik-Programm	5,00
0302	50 Z/Z Umbauanl. o. Teile (engl.)	29,80
0303	Datenseperator f. Superb. Floppy	19,80
0304	Joyst. Bauanl. f. Superb. (engl.)	19,80
0305	Reverse Video	9,80
0306	Cass.-Files	19,80
160	The Fourth Book of Ohio	29,80
8224	Cassette zum Buch Nr. 160	29,80
8228	Joyst. + Software	89,00
8226	Maschinensprache Utilities	19,80
8227	Small Business Progr.	99,00
307	Conversion C2-4P o. C2-8P Comp. i. Disk-Syst. (5 1/4 o. 8 inch),	29,00
8230	Sargon Schach für OSI (C)	119,00
8232	Sargon Schach für OSI (D)	129,00
8231	NEUMON f. C1P u. Superb.	149,00
8233	HEXDOS f. C1PMF (D)	159,00

BASIC Bücher

139	BASIC für blutige Laien	19,80
113	BASIC Handbücher für Anfänger	19,80
121	Microsoft BASIC HB	29,80
122	BASIC für Fortgeschrittene	39,00
31	57 Praktische BASIC Programme	39,00
8057	Computer Games in BASIC	9,80
160	The Fouth Book of OHIO	29,80
255	BASIC/BASIC	39,00
256	Stimulating Simulations	19,80
257	BASIC Computer Programs in Science and Engineering	39,00
260	BASIC Computer Programs	39,00
156	Small Business Programs	29,80
266	Advanced BASIC Applications	39,00
151	Microsoft BASIC	19,80
270	BASIC with Style	39,00

Riesenprogrammsammlung. Ca. 149 Programme in 5 Bänden

8050	BASIC Software, Volume I	99,00
8051	BASIC Software, Volume II	99,00
8052	BASIC Software, Volume III	149,00
8053	BASIC Software, Volume IV	39,00
8054	BASIC Software, Volume V	39,00
8048	BASIC Software, Volume VI	199,00
8049	BASIC Software, Volume VII	159,00
8021	BASIC Software, Volume I–V	425,00

6502 Bücher KIM, AIM, PET, CBM, Challenger

8042	6500 Software Manual	19,80
8043	6500 Hardware Manual	19,80
109	6502 Microcomputer Programm	29,80
118	Programmieren in Ma.-Spr. 6502	98,00
152	Expansion Handbuch 6502	19,80
124	Progr. i. Ma.-Spr. mit CBM	19,80
150	Care and Feeding of the PET	19,80
34	TINY BASIC Handbuch	19,80
1169	The Giant Book of Comp. Projects	39,00
157	The First Book of OHIO Vol I	19,80
158	The Second Book of OHIO	19,80
110	Programmierhandbuch PET	29,80

Best.-Nr.	Titel	Preis/DM
Neu bei Hofacker HAYDEN		
254	The S-100 Handbook	49,00
255	BASIC BASIC	39,00
256	Stimulating Simulations	19,80
257	BASIC Comp. Progr. in Science and Engineering	39,00
258	APL-An Introduction	39,00
259	Creative Progr. for Fun and Profit	29,80
260	BASIC Comp. Progr. f. Business, I	39,00
261	BASIC Comp. Progr. f. Business, 2	39,00
262	Homecomputer can make you rich	19,80
263	Sixty Challang. Problems	19,80
264	The complete 1802 Cookbook	19,80
265	Musical Applications for Micros	79,00
266	Advanced BASIC Appl.	39,00
267	How to profit from your Microc.	39,00
268	Pascal with Style	39,00
269	Cobol with Style	39,00
270	BASIC with Style	39,00
271	BASIC FORTRAN	45,00
272	Z80 and 8080 Assembly Language Programming	39,00
273	Beat the ODDS: Microcomputer Simulations of Casino Games	39,00

ELCOMP Books in Englisch

150	Care a. Feeding of the Comm. PET	19,80
151	8K Microsoft Basic Ref. Manual	19,80
152	Expansion Handb. f. 6502 u. 6800	19,80
153	Mickrocmputer Appl. Notes (Intel)	29,80
154	Complex Sound Gen. w. Microc.	19,80
155	The First Book of 80 US (TRS-80)	19,80
156	Small Business Programs	29,80
157	The First Book of Ohio Scientific	19,80
158	The Second Book of OHIO	19,80
159	The Third Book of OHIO	19,80
160	The Fourth Book of OHIO	29,80
161	The Fifth Book of OHIO	19,80
162	ATARI Games in BASIC	19,80
163	The Periph. Handb.	29,80
164	ATARI Progr. Learning by Poing	29,80

TAB Books

574	Beginner's Guide to Computer Pr.	39,00
752	Computer Programming Handbook	45,00
785	Microprocessor/Microprogramming	35,00
952	Microcomp. Progr. f. Hobbyist	39,00
1000	57 Practical Programs in BASIC	35,00
1015	Beginner's Guide to Microproc.	29,80
1055	The BASIC Cookbook	24,80
1071	Complete Handbook of Robotics	29,80
1085	24 Ready to Run Progr. in BASIC	24,80
1088	Illustrated Dictionary of Microc.	35,00
1095	Programs in Basic fo. Electr. Eng.	19,80
1070	Digital Interfacing	39,00
1141	How to Build your own working Robot PET	29,80
1076	Artifical Intelligence	29,80
1111	How to Design, Build + Program your own working Computer System	29,80
1099	How to Build your own work. 16 Bit Microc.	14,80
1062	The A to Z Book of Comp. Games	29,80
1053	Microprocessor Cookbook	24,80
1045	The Programers Guide to LISP	24,80
1050	The most pop. Subrout. in BASIC	24,80
1169	The Giant Book of Comp. Projects f. 8080/6502	39,00
1187	The Fortran Cookbook	29,80
1203	Handb. of Microproc. Appl.	29,80
1205	PASCAL	35,00
1236	Fiberoptics	29,80
1271	Microcomp. Interfacing	35,00
1275	33 Chall. Comp. Games	29,80

Elektronik Fachbücher

1	Transistor-Berechnungs- und Bauanleitungs-Handbuch	19,80
2	TBB, Band 2	19,80
3	Elektronik im Auto m. Handbuch f. Polizei-Radar	9,80
4	IC-Handbuch (TTL, CMOS, Linear)	19,80
5	IC-Datenbuch	9,80
6	IC-Schaltungssammlung	9,80
7	Elektronikschaltungen zum Basteln	5,00
8	IC-Bauanleitungs-Handbuch	19,80
9	Feldeffektransistoren	5,00
10	Elektronik und Radio, IV	19,80
11	IC-NF-Verstärker	9,80
12	Beispiele integrierter Schaltungen	19,80
13	Hobby-Elektronik-Handbuch	9,80
14	IC-Vergleichsliste, TTL, CMOS (neu)	29,80
15	Optoelektronik-Handbuch	19,80
16	CMOS, Teil 1	19,80
17	CMOS, Teil 2	19,80
18	CMOS, Teil 3	19,80
19	IC-Experimentier-Handbuch	19,80
20	Operationsverstärker	19,80
21	Digitalt.-Grundkurs	19,80
23	Elektronik-Grundkurs	9,80

University Software

Application Programs in Microsoft BASIC (Exidy Sorcerer). 5 Bände mit 105 sehr guten Programmen in Spiralbindung zum Gesamtpreis von 543,00

8600	Small Business	199,00
8601	Education u. Scientific	139,00
8602	Fun u. Games, Volume 1	59,00
8603	Fun u. Games, Volume 2	59,00
8604	Home u. Economics	99,00

Ing. W. HOFACKER GmbH
Tegernseerstr. 18, D-8150 Holzkirchen
Tel.: (0 80 24) 73 31

Lieferung durch den Fach- und Buchhandel od. per NN oder Vorkasse. Postscheck-kto. Mchn 15 944 – 807 oder Euroscheck.
Preise incl. MwSt., zzgl. Porto und NN-Gebühr. Unverbindliche Preisempfehlung. Angebot freibleibend, Zwischenverkauf vorbehalten.

Best.-Nr.	Titel	Preis/DM
NEU	**Neuheiten 1981**	
130	Programme für CBM	19,80
116	16 Bit Microcomputer (400 Seiten)	29,80
35	Der freundliche Computer	29,80
114	Der Microcomputer i. Kleinbetrieb	39,80
120	Anwenderpr. TRS80/Video Genie	29,80
122	BASIC für Fortgeschrittene	39,00
32	ATARI BASIC Handbuch (400 S.)	29,80
137	FORTH Handbuch + Einführung	39,00
136	Der Microcomp. i. Unterrichtsfeld	29,80
139	BASIC für blutige Laien	19,80

ELCOMP Erweiterungsplatinen

(Platinenservice zu den Artikeln in Elcomp) Wir liefern die Platine mit Anleitung und wenn notwendig die zugehörige Software. ELCOMP hat für seine Leser ein neuartiges Erweiterungssystem für alle 6502 Microcomputersysteme entwickelt. Grundlage dieses Konzeptes ist die ELCOMP1 Expansionsplatine. Sie ermöglicht es, jeden 6502-Computer so zu erweitern, daß Sie

1. 4 Steckplätze für Platinen mit Apple Bus und
2. 1 Steckplatz für S44-Bus erhalten.

Die Philosophie dieses Erweiterungssystems liegt darin, daß alle 6502 Systembesitzer Platinen mit dem so weit verbreiteten APPLE II Bus verwenden können. Das Angebot an Erweiterungsplatinen für den APPLE II ist riesig und das kann jetzt jeder 6502-Systembesitzer für sich ausnutzen.

Zum Einstecken in die ELCOMP1 Expansionsplatine haben wir zur Zeit folgende Karten fertiggestellt:

ELCOMP Universal Expansion Boards

Expapsions-Platinen für APPLE II, APPLE II plus und Ohio Scientific Superboard C1P/C4P etc.

604	Universalexperimentierplatine	59,00
605	Ein-/Ausgabe Experimentierpl.	89,00
606	Bus Expansion ELCOMP1	129,00
607	EPROM-Platine f. 2716-Burner	149,00
613	8K RAM-Karte (nur über Bus Expansion anschließbar	69,00
612	32K RAM-Karte dynamisch	169,00
608	Musik-Platine für GI-AZ 8912	89,00
609	EPROM-Karte für 2716	59,00
610	Analog-Digitalwandler-Karte	149,00
611	Rechnerkopplung APPLE II/Superboard, 2 Platinen mit Anleit. u. Softw.	249,00

Für den Microcomputerfreund
Sonderangebote – solange Vorrat reicht.

350	10 Creative Computing Hefte gem.	49,00
351	20 Creative Computing Hefte gem.	69,00
352	9 Byte Magazine Hefte gemischt	29,00
553	AIM-Manual, 6502 Hardware Manual, 6502 Softwaremanual, 2 Programmierkarten, Schaltplan zus.	79,00
354	10 Dr. Dobbs Hefte gemischt	49,00
355	4 6502 User Notes Hefte	29,00
356	8048 Microcomputer Handbuch	5,00

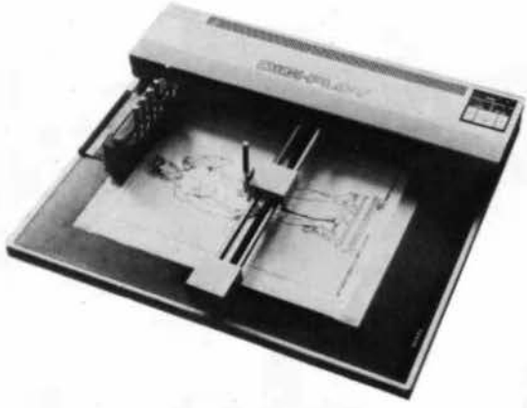
Achtung NEU!

Unser neuer 170seitiger Katalog mit genauen Beschreibungen, Tips, Informationen u. v. a. ist da! Bücher Software, Bausätze, Zeitschriften, Größte Auswahl an Microcomputer-Literatur und Software in Zentral-Europe.
134 Schutzgebühr 2,00 (Vorkasse oder in Briefmarken). Heute noch anfordern. Wird bei einer Bestellung kostenlos mitgeliefert.

Bücher für TRS-80, ZX-80, Video Genie etc.

8029	Z-80 Assemblerhandbuch	29,80
119	Programmieren in Masch.-Spr. Z-80	49,00
111	Progr. mit TRS-80 und Z-80	29,80
155	The First Book of TRS-80	19,80
250	TRS-80 Beginners Programs	29,80
251	TRS-80 Sargon Chess Book	49,00
252	Z-80 Referenz-Karte	5,00
253	How to build a Comp. Controlled Robot (with KIM-1)	35,00
272	Z80 + 8080 Assembly Lang. Progr.	39,00
208	TRS-80 User Journal	14,80

WATANABE GMBH



Manche meinen...

6-Farben-Plotter
müssen teuer sein.

Wir nicht !

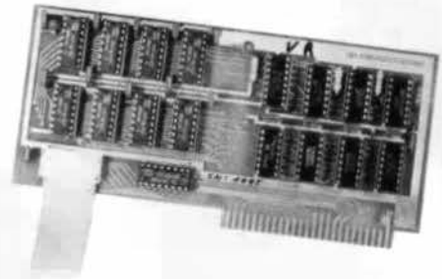
Fordern Sie unverbindlich
ein Informationsangebot an!

WATANABE GMBH
Postfach 1155 · D-8036 Herrsching
Telefon 08152-2526 · Telex 527719

WATANABE GMBH

IBS COMPUTERTECHNIK
Computersysteme · Hard- und Softwareentwicklung

Eurointerfaces: Mehr Leistung für APPLE II und ITT 2020



Bei Ihrem Händler:

AP 1: 16-K-RAM-Karte
Macht aus Ihrem 48-K-Apple einen 64-K-RAM-Rechner, für alle Sprachen und Programme geeignet. Getestet mit INTEGER-BASIC (DOS 3.3) PASCAL, CP/M (Z-80-Karte) und der neuen Sprache LOGO.
Nur für Besitzer des APPLE-PASCAL-Systems nicht erforderlich.

AP 2: Serien-Schnittstelle mit 6551.

AP 3: PS-PALCODER.
Eine PAL-Karte neuester Entwicklung, bei der rot = rot und blau = blau ist. Für alle, die Farbe wollen und nicht nur bunte Bilder.

AP 4: Parallel-Schnittstelle mit 6522.

AP 5: Relais-Karte 4- und 8fach.

AP 6: Optokoppler-Karte 4- und 8fach.

AP 7: Digitalvoltmeter.

AP 8: 16-K-EPROM-KARTE.

In Vorbereitung:

Fragen Sie Ihren Händler nach diesen Interfaces, vielleicht zeigt er Ihnen schon jetzt:

AP 9: Multi-Interface.
Darauf haben viele gewartet, endlich mehr Platz durch Uhr – A/D-Wandler. Serieller und paralleler Schnittstelle auf einer Karte.

AP 10: 64-K-RAM und 6809-Karte.
Für den Super-Apple II, endlich 128 K und Doppelprozessorsystem.
Eine ungewöhnliche Leistung für Mikrosoftware-Evolution.

AP 11: Intelligenter Floppy Controller.
Vier Laufwerke an einem Slot-Platz in Apple und noch mehr Massenspeicher-Shugart-Laufwerke anschließbar.

AP 12: 80 Zeichen Terminal-Karte
Zeichen und Zeilenzahl programmierbar. Zeichensatz fest oder im EPROM oder beides mit 16-K-RAM und 12 Bit Tiefe.

AP 13: Einfaches Druckerinterface.
Damit Sie Ihren Geldbeutel schonen können.

Weiter geht es mit vielen interessanten Interfaces, die Sie bald bei Ihrem Händler kennenlernen können – oder in der APPLE USER GROUP EUROPE (AUGE). Ihr Händler ist Ihr Partner in allen Fragen zu Ihrem APPLE – er freut sich auf Ihren Besuch. Er ist unser Partner, damit Sie mehr von Ihrem Apple haben.

Alle Karten stammen aus deutscher Entwicklung und deutscher Fertigung.
Ausführliche deutsche Benutzeranleitung, hohes Maß an Sicherheit und 12 Monate Garantie:

IBS Interfaceskarten erhalten Sie nur bei Ihrem Computerfachhändler.

Ingenieurbüro Specovius

Artur-Ladebeck-Straße 143, 4800 Bielefeld 14

Der **SM-KIT** ist eine Sammlung von Programmier- und Testhilfen für BASIC-Programmierer.

HELP: Bei SM-KIT wird bei Fehlermeldungen **automatisch** die Fehlerzeile auf den Bildschirm ausgegeben und der Cursor auf die aktuelle Interpreterspalte gestellt.
AUTO: Beim Schreiben der Zeilennummer in die erste Spalte wird automatisch die nächste Zeilennummer vorgegeben.
Steuerung der Ausgaben auf den Bildschirm: Die Ausgaben: FIND, DUMP, TRACE, DIRECTORY laufen nur, solange RETURN gedrückt ist, stoppen beim Loslassen von RETURN und setzen bei erneutem Niederdrücken automatisch fort.
Ausgabe auf Drucker oder Floppy: Durch ein einziges Kommando Ausgabe auf Standarddrucker (4=3022) oder nach OPEN durch das gleiche Kommando auf beliebige Drucker oder auf Floppy.
HARDCOPY: Ausgabe eines beliebigen Bildschirmzeilenbereichs auf Drucker oder Floppy.
FIND: Findet Variablen exakt (bei Suche nach A wird nicht AB gefunden!). Findet alle Integer- bzw. Stringvariablen bzw. Felder durch Vorgabe von $ oder $(oder % oder %(. Findet Zahlen exakt (bei 50 wird nicht GOTO 500 gefunden) oder mit Joker (50! findet 500, 501 ... 509). Kann Komma, Doppelpunkt und Anführungszeichen suchen.
NUMBER: Die Numerierungsfunktion kann erstens auf bestimmte Zeilenbereiche beschränkt werden und kann zweitens **Programmblöcke** in ihrer statischen Reihenfolge **gegeneinander austauschen!**
DUMP gibt einfache Gleitkomma-, Integer- und Stringvariablen getrennt aus. Bei **Feldvariablen** kann entweder die Dimensionierung aller Felder oder der **Inhalt einzelner Felder** unter Angabe der Startindizes ausgegeben werden.
TRACE zeigt einzelne Statements (nicht nur Zeilennummern) und kann zusätzlich **Variableninhalte** mit ausgeben. Diese Variablen können jederzeit geändert werden. Zeitsparend und gezielt wird TRACE durch Vorgabe der Zeilennummer zu Beginn oder während des Programmablaufes angewendet. Der TRACE kann durch die RETURN-Taste sehr elegant gesteuert werden, so daß fortlaufender TRACE und SINGLE-STEP in einer Funktion enthalten sind.
Floppy-Kommandos: Hier dürfen im Gegensatz zu DOS-SUPPORT **Variable** verwendet werden, was z.B. bei COPY oder SCRATCH hilfreich sein kann.
LOAD beinhaltet durch Angabe eines Zusatzparameters APPEND. **Maschinenprogramme** können **ohne Veränderung** der Variablenzeiger (Basic bleibt warm) geladen werden. Das Ablegen **verschiebbarer** Maschinenprogramme in **beliebige** Speicherbereiche ist unter gleicher Bedingung möglich.
MERGE bietet im Gegensatz zu APPEND die Möglichkeit, wirklich beliebige Zeilen von der Diskette in den Arbeitsspeicher zu übernehmen.
SAVE macht automatisch VERIFY, kann **Teile eines BASIC-Programms** und beliebige Speicherbereiche (Maschinenprogramme) abspeichern.

SM-KIT
für Commodore-Computer

Profiwerkzeug
für Programmierer

(4-K-ROM mit ausführlicher deutscher Anleitung)

Zu beziehen: **Deutschland:** Commodore-Fachhändler · **Schweiz:** Instant-Soft AG., CH-5507 Mellingen · **USA:** AB-Computers, Montgomeryville, PA 18936
Eine Entwicklung der **SM Softwareverbund-Microcomputer GmbH, Scherbaumstraße 29, 8000 München 83**

Unser Kompakt-Computer Z-89 spart der Industrie in vielfacher Weise Geld.

1. Mit dem Kompakt-Computer Z-89 besitzen Sie ein anspruchsvolles, absolut Industrie-fähiges System.

2. Der Z-89 ist robust und zuverlässig: Gestochen scharfe Terminalschrift, Tastatur für professionellen Büroeinsatz, fundierte Dokumentation für Hard- und Software.

3. Die äußerst günstige Preis/Leistungs-Relation ist sprichwörtlich.

4. Unser Z-89 kann viele Aufgaben selbständig erledigen – er kann aber auch mit Großcomputern zusammenarbeiten (DfÜ-fähig).

Summa summarum: Unser Kompakt-Computer spart Ihnen in vielfacher Weise Geld.

Z-89 Kurzprofil:
● 24 + 1 Zeile à 80 Zeichen ● 2 x Z80 im Grundsystem ● Ausbaufähig bis 64kB RAM ● Extern bis 20 Megabyte ● Datenbus zugänglich ● Serielle und parallele I/O's ● Feingrafik (512 x 256 Punkte) – auf Wunsch ● PROM – programmierfähig ● IBM – 3740 – kompatibel

Grundsystem schon ab DM **3.956,–**

HEATH
HEATH Zenith Daten-Systeme

HEATH GmbH
Ausstellungs- und Service-Zentrum
Robert-Bosch-Straße 32-38
Postfach 102060
Telefon 06103/3808 · Telex 0417986
6072 Dreieich-Sprendlingen

COUPON Bitte senden Sie mir ausführliche Informationen mc 5/81
☐ Computer ☐ Speicher ☐ Drucker ☐ Zubehör
Name
Straße
PLZ/Ort

Mikroprozessor-Entwicklungen mit eigenem Prozeßrechner.

Nutzen Sie die Leistungsfähigkeit Ihres Prozeßrechners für Ihre Software-Entwicklung und Simulation.

Durch die Unabhängigkeit arbeiten Sie wirtschaftlicher und können sich jederzeit an neue Aufgaben anpassen. Ihr Prozeßrechner, kombiniert mit den von uns gelieferten Cross-Assemblern, Simulatoren und Cross-Compilern, kann für über 50 Mikroprozessoren eingesetzt werden.

Ganz gleich ob Sie im single-user oder multi-user System arbeiten. Boston Systems hat die passende Entwickler-Station. Die Spezialisten der RSE helfen Ihnen gerne bei der wirtschaftlichen Lösung Ihrer Mikroprozessor-Entwicklungen.

Rohde & Schwarz Engineering and Sales GmbH
Graf-Zeppelin-Straße 18
5000 Köln 90
Telefon (0 22 03) *49-1
Telex 8 873 288 (rse d)

ROHDE & SCHWARZ
RSE
ENGINEERING AND SALES GMBH

TRS-80 COMPUTERSYSTEME

Ab DM 100,– Auftragswert erfolgt die Lieferung porto- und verpackungsfrei. Lieferung per Nachnahme oder Vorauszahlung. Alle Preise incl. Mwst.!

HARTWARE für Modell I:
TRS-80 Computer ohne Monitor u. Recorder 4K Lev. I **DM 895,–.**
Expansion Interface **DM 895,–.**
Mini Disk 1. Einheit mit DOS, Manual und 4er Kabel **DM 1250,–.**
Mini Disk 2.-4. Einheit **DM 1150,–.**

8 Zoll Disk Laufwerk (Double Density) **DM 1795,–.**
Quick Printer II **DM 645,–.**
LINE PRINTER III **DM 4150,–.**
Level II Erweiterung incl. Einbau **DM 299,–.**
16K RAM Erweiterung incl. Einbau **DM 295,–.**
RS 232c Schnittstelle **DM 289,–.**
10er Tastatur incl. Einbau **DM 129,–.**
Umrüstung auf Gross-/Kleinschreibung incl. Einbau **DM 118,–.**
10er Pak Mini Disketten BASF Double Density **DM 95,–.**

NEUHEITEN:
LINE PRINTER I **DM 1995,–.**
LINE PRINTER VI **DM 2395,–.**
DAISY WHEEL Printer II **4195,–.**
PLOTTER/PRINTER für Endlos Papier mit A4 Breite **DM 2695,–.**
Grafik Erweiterung zum Ansprechen von 192x384 Dots **DM 445,–.**
Data Separator für Expansion Interface **DM 99,50.**
Double Density Nachrüstung für Exp. Interface **DM 465,–.**
Adapter zum Anschluss von 8" Laufwerken an Mod. I **DM 285,–.**
Analog / Digital Wandler mit 8 Kanälen **DM 395,–.**
MINI DISK II mit 80 Tracks (204K Byte) **DM 1450,–.**
TRS-80 POCKET Computer **DM 448,–.**

Software Hits für TRS-80 Computer:

	Tape	Disk
APL-80 Interpreter	DM 99,50	149,50
Level III BASIC	DM 149,–	149,–
BASIC-Compiler	DM 225,–	450,–
COBOL-Compiler	DM –	695,–
FORTRAN-Compiler	DM –	189,–
MACRO-Assembler	DM 129,–	189,–
Editor/Assembler	DM 99,50	109,50
muMATH/muSIMP	DM –	195,–
PASCAL Compiler	DM 99,50	395,–
SCRIPSIT Textverarbeitung	DM 149,50	208,50
Geschäftsadress-programm	DM 42,50	148,–
Inventory Control System (deutsch)	DM –	139,–
Wärmebedarfs-berechnung	DM –	169,–
Microchess	DM 42,50	42,50
SARGON II	DM 89,–	89,–

Modell II Software:
BASIC-Compiler incl. Macro-Assembler, Loader u.a. **DM 995,–.**
FORTRAN-Compiler incl. Editor, Linking-Loader u.a. **DM 595,–.**
MACRO-Assembler incl. Cross-Reference, Editor u.a. **DM 595,–.**
COBOL-Compiler incl. Editor, Loader u.a. **DM 595,–.**
PASCAL-Compiler (CP/M-Version) **DM 795,–.**
Finanzbuchhaltung (bis zu 3000 Buchungen/ Mon.) **DM 2450,–.**
Inventory Control System (deutsch) **DM 598,–.**

NEU!

SCRIPSIT	Textverarbeitungssystem das Maßstäbe setzt **DM 995,–.**
PROFILE	Database Management System. Arbeitet zusammen mit SCRIPSIT **DM 995,–.**
MEMSORT	Sortiert 1000 Strings in ca. 5 Sekunden **DM 169,–.**
DISKSORT	Sortiert in Rekordzeit RANDOM-Dateien **DM 495,–.**
IBM 3741	Liest, formatiert und schreibt Disketten im IBM 3741 Format **DM 395,–.**

TANDY TRS-80
Computer mit 4K RAM Speicher und Level II BASIC als Komplettsystem mit Videomonitor, (grüne Bildröhre), Cassettenrecorder zum Speichern von Daten und Programmen, ausführlichem Bedienungshandbuch in Deutsch für nur **DM 1395,–.**

TANDY TRS-80 Computer w.o. aber mit 16K RAM grossem Programmspeicher und zusätzlicher nummerischer Tastatur und Gross/Kleinschreibung für nur **DM 1795,–.**

TANDY TRS-80 MODELL II
Mit 64K RAM Speicher und eingebauter 8" Floppy Disk. Umfangreiche Betriebssoftware incl. BASIC Level III **DM 9950,–.**

FÜSSNER COMPUTERSYSTEME GMBH

4430 Steinfurt, Markt 17, Tel. 02551/2426

4440 Rheine, Hörstkamp 7, Tel. 05971/12539

Intelligenz

Zeichnen + Drucken + Digitalisieren

Wo bisher für jede dieser Aufgaben ein eigenes teures Peripheriegerät erforderlich war, setzt adcomp hier ein Zeichen der Zeit:

Printer-Plotter X 80 SP

Mit hoher Intelligenz für minimalen Programmieraufwand. Für jeden Rechner und alle Schnittstellen.

Plotten und Drucken im Format DIN A4 in allen Richtungen. Zur schnellen Ausgabe von Tabellen, Funktionen, Zeichnungen und Bildern.

adcomp X 80 SP. Technik die überzeugt. Sicherer, wartungsfreier Betrieb auch in rauher Umgebung. Einfach in der Bedienung. Günstiges Preis-/Leistungsverhältnis. Deutsche Wertarbeit. Intelligent und zuverlässig.

adcomp
Datensysteme GmbH
Horemansstraße 8
8000 München 19
Telefon 089/19 40 19
Telex 05 216 271

adcomp

Ihr Partner für intelligente Peripherie

Detaillierte Daten nennen wir Ihnen gern auf Anfrage.

---

computer shop

bietet auch Zusatzkarten für die Personal Computer Apple, ITT 2020 sowie AIM-65-PC 100-SYKO 100 an.

MC 5-81

Fordern Sie eine aktuelle Preisliste und Beschreibung an!

Für Profis und OEMs
Info zum:

Rechnersystem CS-2000

Unsere neue Adresse finden Sie:
auf der 3. Umschlagseite (b. Anforderungskarten)

DATA BECKER DATA BECKER DATA BECKER DATA BECKER DATA BECKER DATA

DATA BECKER DATA BECKER

1. Düsseldorfer Computerwochen

Vom 14.-27. Mai, werktags 10-18 und samstags 10-14 Uhr

Unsere große Ausstellung mit vielen aktuellen Neuheiten vom Internationalen Computermarkt. Wir zeigen Kleincomputer für alle Anwendungsbereiche, vielseitige Peripheriegeräte, eine umfassende Zubehörauswahl, eine riesige Softwarepalette von der Systemroutine bis zur ausgereiften kommerziellen Lösung für zahlreiche Branchen, sowie Europas größte Auswahl an EDV-Literatur mit vielen neuen Titeln und Publikationen. Wir führen mit Ihnen ein Expertengespräch oder machen Sie mit der EDV vertraut. Wir präsentieren Ihnen unsere Leasing- und Finanzierungsmöglichkeiten und unterbreiten Ihnen ein Leistungsfähiges Angebot für Ihre bisherige Anlage. Oder kurz gesagt:

Das komplette EDV-Angebot

DATA BECKER

DATA BECKER GMBH im Hause Auto Becker
Merowinger Straße 30 · 4000 Düsseldorf
Telefon (0211) 312085 · Telex 08582874

Unser Super-Info DATA BECKER NEWS 3/81 erscheint am 15. Mai mit über 60 Seiten aktueller Angebote und einem brandheißen Bericht aus den USA über die Neuheiten der NCC. - Sofort anfordern gegen 3,— DM in Briefmarken.

DATA BECKER DATA BECKER DATA BECKER DATA BECKER DATA BECKER DATA

Franzis-Sonderhefte
für alle Mikrocomputer-Interessierte

Welchen Weg der Mikrocomputer gehen wird, läßt sich noch nicht vorherbestimmen. Vor allem im Anwendungsbereich steht die Entwicklung erst am Anfang. Eines aber ist absehbar, jeder der auch nur am Rande mit Elektronik zu tun hat, wird davon berührt werden bzw. ist heute schon fest damit verbunden.
Mit diesen Sonderheften wollen wir jeden Berufs- oder Hobby-Elektroniker aktuell über den neuesten Stand informieren. Welches Heft für Sie als Informationsquelle in Frage kommt, können Sie aus nachstehender Tabelle ersehen.

Programme für Taschenrechner (HP- und TI-Serien), für Basic-Computer und für die Mikroprozessoren Z-80 und 6502 auch in Maschinensprache

Zahlreiche neue Beiträge.

Für alle Computerbesitzer und solche, die es werden wollen

80 Seiten, DM 14.80

Hardwareorientierte Einführung mit µP 2650.

Sammlung der in der ELO bereits erschienenen gleichlautenden Beitragsreihe.

Anfänger

144 Seiten, DM 18.–

Hard- und Software-Grundlagen.

Gesammelte Beiträge der in der FUNKSCHAU bereits erschienenen gleichnamigen Reihe.

Anfänger

68 Seiten, DM 9.50

Grundlagen, einfache Basic- und Maschinenprogramme.

Überwiegend Beiträge, die bisher nirgendwo erschienen sind.

Anfänger und Fortgeschrittene

136 Seiten, DM 19.–

Basic- und Maschinenprogramme, Hardware-Tips.

Ausschließlich exklusive Beiträge.

Fortgeschrittene

80 Seiten, DM 14.–

Programmierhilfen, Markttendenzen, 16-Bit-Prozessoren, Adressen von Herstellern.

Überwiegend Beiträge aus der ELEKTRONIK.

Industrielle Anwender

136 Seiten, DM 18.–

Programme in Basic, Maschinensprache, Pascal und für Taschenrechner der HP- und TI-Serien.

Die Beiträge sind sonst nirgendwo erschienen.

Für alle Computer-Besitzer und solche, die es werden wollen.

80 Seiten, DM 15.60

Grundlagen, Technik, praxisnahe Anwendungen.

Überwiegend bewährte Beiträge aus der ELEKTRONIK.

Industrielle Anwender, die sich mit dem Einsatz von automatischen Meßsystemen beschäftigen.

84 Seiten, DM 19.–

Grundlagen, Programmiertechnik, Unterschiede zu anderen Programmiersprachen, Beschreibung von Systemen.

Überarbeitete Beiträge aus der ELEKTRONIK.

Mikrocomputer-Entwickler und -Anwender

64 Seiten, DM 18.50

Hier erhalten Sie diese Sonderhefte:

Bei allen Bahnhofsbuchhandlungen, größeren Buchhandlungen und Elektronik-Bauteilehändlern oder gegen Vorauszahlung direkt vom Franzis-Verlag.

Wir bitten Sie, in diesem Fall als Bestellung den genannten Betrag plus 1.50 DM Porto auf unser Postscheckkonto München Nr. 813 75-809 mit genauer Nennung des jeweiligen Titels zu überweisen oder einen Scheck über diese Summe einzusenden.

Franzis-Verlag
Karlstraße 37, 8000 München 2
Telefon (0 89) 51 17-2 39.

Die Hefte erhalten Sie in der Schweiz auch beim Verlag Thali AG, CH-6285 Hitzkirch
und in Österreich beim Fachbuch Center Erb, A-1061 Wien, Amerlingstraße 1

Babylon's Datenverwirrung

Selbst wenn alle Computer die gleiche (Programmier-)Sprache sprechen würden, könnten sie untereinander noch längst keine Daten austauschen. Die Hersteller verwenden nämlich die unterschiedlichsten Datenformate an seriellen Schnittstellen. Wir haben uns die Mühe gemacht, hier die wichtigsten zusammenzustellen; vorangestellt sei eine kurze Erläuterung der Zeichendarstellung in Computern.

Computer arbeiten heutzutage in rein binärer Logik, sie können von Haus aus also nur zwei Zeichen darstellen – Null und Eins („0" und „1"). Da man Computer aber auch zur Darstellung aller Ziffern unseres Dezimalsystems sowie aller Buchstaben und Satzzeichen bewegen möchte, hat man sich einige Codes einfallen lassen, um Gruppen von mehreren Bits (nicht unbedingt immer ein ganzes Byte) zu einem Zeichen zusammenfassen.

n Bits ergeben 2^n Zeichen

Man kann sich leicht überlegen, daß es zum Beispiel bei einer Gruppe von drei Bits, die jeweils nur zwei Zustände Null und Eins einnehmen können, $2 \times 2 \times 2 = 8$ Kombinationsmöglichkeiten gibt. Definiert man jede dieser Bitkombinationen als Repräsentant für ein Zeichen, zum Beispiel für eine dezimale Ziffer, so könnte unser 3-Bit-Mini-Code so aussehen:

000	„0"
001	„1"
010	„2"
011	„3"
100	„4"
101	„5"
110	„6"
111	„7"

Diese 2^3 Zeichen (0...7) reichen für unser Dezimal-Zahlensystem natürlich noch nicht aus. Immerhin gibt es den 3-Bit-Code in der Praxis aber tatsächlich; er nennt sich „Oktalcode", weil er acht Zeichen umfaßt. Dementsprechend gibt es auch ein Oktal-Zahlensystem, das – im Gegensatz zum Dezimalsystem – nur acht Ziffern umfaßt. Die dezimale 7 ist auch oktal noch 7, die dezimale 8 ist im Oktalsystem aber 10, weil – so die Definition aller Stellenwertsysteme – immer dann links eine Stelle hinzugenommen wird, wenn der Ziffernvorrat für eine bestimmte Zahl nicht mehr ausreicht.

Das Hexadezimal-System

Um alle dezimalen Ziffern von 0...9 darstellen zu können, benötigt man also vier Bits. Umgekehrt ergeben vier Bits aber 2^4 mögliche Zeichen, also 16. Ein in Gruppen von je vier Bits aufgebautes Zahlensystem nennt man deshalb Sedezimal-System, üblicherweise – wenn auch linguistisch nicht hundertprozentig korrekt – Hexadezimal-System.
Das Hexadezimal-Zahlensystem umfaßt nach allgemeiner (willkürlicher) Vereinbarung folgende sechzehn Zeichen bzw. folgende „Ziffern":

0000	„0"	1000	„8"
0001	„1"	1001	„9"
0010	„2"	1010	„A"
0011	„3"	1011	„B"
0100	„4"	1100	„C"
0101	„5"	1101	„D"
0110	„6"	1110	„E"
0111	„7"	1111	„F"

Die links hingeschriebenen binären Ziffern besitzen dabei Wertigkeiten entsprechend den Zweierpotenzen 2^3, 2^2, 2^1 und 2^0. Für die Hex-Ziffer „C" ergibt sich deshalb eine Wertigkeit der binären Darstellung von $1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 =$ (dezimal) 12. Die „Ziffern" A...F entsprechen also den dezimalen Zahlenwerten 10...15.
Die dezimale Zahl 16 läßt sich hexadezimal als 10 schreiben – denn auch hier wird ja genau dann links eine Eins hinzugeschrieben, wenn der systemgegebene Ziffernvorrat gerade nicht mehr zur Darstellung einer bestimmten Zahl ausreicht.
Dementsprechend lassen sich hexadezimale Zahlen auch mehrstellig aneinanderschreiben, z. B. als FEC0 oder 03A4 usw. Bei Ausdrücken wie 200 sollte man aufpassen, in welchem Zahlensystem sie zu verstehen sind – hexadezimal 200 entspricht einem dezimalen Wert von 512, was man leicht durch Zuordnung einer Hex-Ziffer zu einer 4-Bit-Gruppe und anschließender Aufsummierung der Bitwertigkeiten feststellen kann.
Die acht Bits eines Bytes bzw. ein Datenwort bei 8-Bit-Mikrocomputern werden üblicherweise in Form zweier, die 16-Bit-Adresse in Form von vier Hex-Ziffern notiert. So bedeutet der „Hex-Dump"

```
0200 1A 4B 00 30
```

etwa, daß an der Hex-Adresse 0200 das Datenbyte 1A, bei 0201 das Byte 4B usw. steht. Bei solchen Hex-Dumps sollte man sich keine langen Gedanken darüber machen, was die Werte jeweils dezimal bedeuten – das interessiert im Grunde niemanden, und den Computer schon gar nicht. Er interpretiert die Hex-Ziffern jeweils als Folge von vier Bits.

Was bedeutet ASCII?

Einen kleinen Schritt sind wir bei unserem Bemühen, mit Nullen und Einsen und ihren Kombinationsmöglichkeiten Buchstaben zu codieren, schon vorangekommen, gelang es uns doch immerhin schon, sechzehn unterschiedliche Zeichen mit Ja-Nein-Entscheidungen darzustellen.
Wenn wir noch einen Schritt weitergehen, so lassen sich auch alle Groß- und Kleinbuchstaben sowie die Satzzeichen codieren. Man kann sich leicht ausrechnen, daß dafür auch fünf oder sechs Bits nicht ausreichen würden – nötig sind sieben Bits, die $2^7 = 128$ Kombinationsmöglichkeiten bieten. Welches Zeichen welcher Bitkombination zugeordnet werden soll, wurde zum Glück international weitgehend

festgelegt. Dafür gibt es die sog. ISO-7-Bit-Norm, die den einzelnen Staaten auch einen gewissen Freiraum für die Implementierung nationaler Sonderzeichen (so etwa der deutschen Umlaute ä, ö, ü) läßt. Die „Urform" ist allerdings die ASCII-Norm (American Standard Code for Information Interchange), die wir in *Bild 1* dargestellt haben [1].

Da man in der Praxis ohnehin nicht in binärer Zahlendarstellung arbeitet, sondern entweder mit den dezimalen oder hexadezimalen Äquivalenten, enthält Bild 1 nicht mehr die den einzelnen ASCII-Zeichen entsprechenden Bitkombinationen, sondern ihre Hex- und Dezimal-Entsprechungen.

Der 7-Bit-Code beinhaltet einige Überlegungen seiner Entwickler, die dem Anwender das Leben manchmal leichter, manchmal schwerer machen. So ist es relativ leicht, aus den Ziffern „0" bis „9", die hexadezimal als 30 bis 39 codiert sind, ihre binäre Wertigkeit abzuleiten, indem man einfach die höherwertigen drei der sieben Bits wegläßt; demonstriert sei das an der „5": Diese Ziffer findet sich in unserer ASCII-Tabelle unter dem Hex-Wert 35, der binär als 0110101 geschrieben werden kann. Läßt man die ersten drei Bits weg, so bleibt 0101 übrig, was binär genau die Wertigkeit 5 besitzt.

Etwas umständlicher wird es beim Decodieren der Hex-Ziffern A...F, die in ASCII ja ganz normale Buchstaben und nicht etwa Ziffern für hexadezimale Mathematik darstellen. Sie müssen getrennt verarbeitet werden, indem man vom ASCII-Hex-Äquivalent (41...46) 37 subtrahiert, so daß die Hex-Werte A...F herauskommen.

Man sollte immer genau auseinanderhalten, daß es die Buchstaben (im Sinn von Text) gibt, die in ASCII ab hex 41 codiert sind, und Hex-Ziffern, die ebenfalls wie Buchstaben aussehen, aber binäre Wertigkeiten repräsentieren.

Da Mikrocomputer heute üblicherweise mit einer Wortbreite von acht Bits (= 1 Byte) arbeiten, bliebe beim 7-Bit-Code immer ein Bit „übrig". Dieses (höchstwertige) Bit in einem Byte verwendet man aber oft für durchaus nicht Unnützes: Entweder belegt man es mit der zur Fehlererkennung bei der Datenübertragung dienenden Parität, die eine auf 1 Bit reduzierte Prüfsumme der sieben ASCII-Bits darstellt, oder man schaltet damit z. B. den Bildschirm-Zeichensatz auf Grafik um. Manche Bildschirm-Datensichtgeräte und Drucker ignorieren das höchstwertige der sieben ASCII-Bits; sie verwerten nur sechs Bits und setzen das siebte intern gleich Null. Der darstellbare Zeichensatz reduziert sich dadurch auf die ASCII-Hex-Gruppen 40...5F und 20...3F. Dadurch fallen erstens alle Kleinbuchstaben und zweitens alle ASCII-Steuerzeichen (Gruppe 00...1F) weg.

hex	dez.	ASCII	hex	dez.	ASCII	hex	dez.	ASCII	hex	dez.	ASCII
00	0	NUL	20	32	Space	40	64	@	60	96	\`
01	1	SOH	21	33	!	41	65	A	61	97	a
02	2	STX	22	34	"	42	66	B	62	98	b
03	3	ETX	23	35	#	43	67	C	63	99	c
04	4	EOT	24	36	$	44	68	D	64	100	d
05	5	ENQ	25	37	%	45	69	E	65	101	e
06	6	ACK	26	38	&	46	70	F	66	102	f
07	7	BEL	27	39	'	47	71	G	67	103	g
08	8	BS	28	40	(	48	72	H	68	104	h
09	9	HT	29	41	)	49	73	I	69	105	i
0A	10	LF	2A	42	*	4A	74	J	6A	106	j
0B	11	VT	2B	43	+	4B	75	K	6B	107	k
0C	12	FF	2C	44	,	4C	76	L	6C	108	l
0D	13	CR	2D	45	–	4D	77	M	6D	109	m
0E	14	SO	2E	46	.	4E	78	N	6E	110	n
0F	15	SI	2F	47	/	4F	79	O	6F	111	o
10	16	DLE	30	48	0	50	80	P	70	112	p
11	17	DC1	31	49	1	51	81	Q	71	113	q
12	18	DC2	32	50	2	52	82	R	72	114	r
13	19	DC3	33	51	3	53	83	S	73	115	s
14	20	DC4	34	52	4	54	84	T	74	116	t
15	21	NAK	35	53	5	55	85	U	75	117	u
16	22	SYN	36	54	6	56	86	V	76	118	v
17	23	ETB	37	55	7	57	87	W	77	119	w
18	24	CAN	38	56	8	58	88	X	78	120	x
19	25	EM	39	57	9	59	89	Y	79	121	y
1A	26	SUB	3A	58	:	5A	90	Z	7A	122	z
1B	27	ESC	3B	59	;	5B	91	[	7B	123	{
1C	28	FS	3C	60	<	5C	92	\	7C	124	|
1D	29	GS	3D	61	=	5D	93	]	7D	125	}
1E	30	RS	3E	62	>	5E	94	↑	7E	126	~
1F	31	US	3F	63	? ▶	5F	95	←	7F	127	DEL

Bedeutung der wichtigsten Steuerzeichen:
NUL = ohne Wirkung, SOH = start of header, STX = start of text, ETX = end of text, EOT = end of transmission, ENQ = enquiry (Aufforderung der Gegenstation zum Senden), ACK = acknowledge (Bestätigung, Rückmeldung), DLE = data link escape (Umschalten auf eine andere Steuerzeichengruppe), NAK = negative acknowledge, SYN = Synchronisationszeichen, ETB = end of transmission block, VT = vertical tabulating (Cursor nach oben), HT = horizontal tab. Cursor nach rechts), BS = Back Space (Cursor nach links), LF = Line feed (Cursor nach unten), BEL = bell (akustisches Zeichen, Klingel), CR = Carriage Return, FS = field seperator, GS = group seperator, US = unit seperator, Space = Leerraum, DEL = Delete/Rub out.

Bild 1. Das ist der ASCII-Zeichensatz mit seinen dezimalen und hexadezimalen Code-Äquivalenten (aus: Basic für Mikrocomputer, Franzis-Verlag)

Nur der Vollständigkeit halber...

Nur der Vollständigkeit halber sei noch ein Code erwähnt, der in der Mikrocomputer-Technik eigentlich kaum noch eine Rolle spielt, mit dem aber üblicherweise Fernschreiber noch arbeiten – und Fernschreiber können ja als Billigst-Drucker für Computer dienen.

Der Baudot-Code arbeitet mit nur fünf Bits. Jetzt werden Sie sagen, wie will man mit fünf Bits alle Buchstaben, Ziffern und Satzzeichen darstellen – 2^5 ergibt ja nur 32 Zeichen?! Die Beschränkung auf fünf Bits wurde mit zwei Maßnahmen erreicht. Erstens verzichtet man beim Baudot-Code auf Groß- und Kleinschreibung; ein Fernschreiber druckt entweder nur klein (das ist die Regel) oder nur groß (solche gibt es auch). Aber auch das würde noch nicht ausreichen, um mit nur fünf Bits auszukommen.

Deshalb führte man zwei „Zeichenebenen" ein – eine Buchstabenebene, die alle Buchstaben von A bis Z enthält, und eine Ziffernebene, die alle Ziffern und Satzzeichen enthält. Das Umschalten zwischen beiden Ebenen geschieht mit besonderen Steuerzeichen (BU und ZI). Manche Zeichen existieren in beiden Ebenen, so z. B. Wagenrücklauf

(CR), Zeilenvorschub (LF) und Zwischenraum (Space) sowie natürlich die beiden Ebenen-Umschaltzeichen.
Auf diesen Code gehen wir hier nicht näher ein, da er im Zusammenhang mit Fernschreiber-Ausgabe-Programmen ohnehin noch öfter auftauchen wird. Ähnliches gilt für den vorwiegend von IBM verwendeten EBCDIC-Code, der in elektrischen Schreibmaschinen und Kassettenaufzeichnungsgeräten angewandt wird, aber sonst möglichst gemieden werden sollte. Man ist gut beraten, bei Eigen- und Neuentwicklungen möglichst ausschließlich ASCII einzusetzen.

Sieben Bits auf zwei Drähten

Wenn man von solchen Techniken wie IEC-Bus und Handshake-Parallelschnittstelle absieht, erfolgt die Übertragung der einzelnen Bits der Zeichen nacheinander auf einer einzigen Leitung mit Erde als Bezugspotential bzw. mit einem Leitungspaar. Den dabei auftretenden zwei Spannungen für „ein" und „aus" bzw. „High" und „Low" lassen sich die binären Wertigkeiten 1 und 0 zuordnen.
Normalerweise weiß der Empfänger aber nicht genau, wann welches Bit mit welcher Wertigkeit über die Leitung gesendet wird. Um dieses Problem der asynchronen Datenübertragung zu umgehen, definiert man, daß die Leitung im Ruhezustand (kein Zeichen wird übertragen) auf 1-Potential liegt. Sobald ein Zeichen gesendet wird, teilt der Sender dem Empfänger dies dadurch mit, daß er die Leitung kurz auf Null-Potential zieht. Dadurch erkennt der Empfänger den Zeichenanfang auch dann richtig, wenn das erste zu übertragende Bit auf 1-Potential liegt und sich damit nicht vom Ruhezustand der Leitung unterscheidet; den kurzen 0-Impuls bezeichnet man als „Startbit".
Dann folgen die sieben ASCII-Bits nacheinander, und zwar das niederwertigste zuerst, gefolgt von einem achten Bit, das sich aus der 8-Bit-Struktur von Mikrocomputern ergibt und entweder konstant auf Null liegt, die Parität beinhaltet (s. o.), oder auch konstant auf Eins liegt.
Um sicherzustellen, daß beim Startbit des nächsten Zeichens auf jeden Fall ein 1-0-Übergang auftritt, macht der Sender zwischen den einzelnen Zeichen jeweils eine kurze Pause, die er mit der Ruhelage-Polarität (1) auffüllt. Diese Pause nennt man „Stopbit".
Wie lange ein Bit auf der Leitung andauert, hängt natürlich von der Übertragungsgeschwindigkeit ab. Bei ASCII sind 110 Bit/s, 150 Bit/s und Vielfache von 300 Bit/s üblich; 300 Bit/s ist z. B. bei akustisch gekoppelten Telefon-Modems ein Standardwert. Bei 110 Bit/s werden jedem Zeichen zwei Stopbits nachgestellt, bei allen anderen Geschwindigkeiten nur eines.
Die Übertragung des Baudot-Codes erfolgt ganz ähnlich; gesendet werden ein Start-, fünf Daten- und 1,5 Stopbits. (Natürlich gibt es keine halben Bits; „1,5" bezieht sich lediglich auf die Mindest-Zeitdauer!)

Die 20-mA-Stromschleife

Ein weitverbreitetes Verfahren, diese graue Theorie in die Praxis umzusetzen, ist, folgende Beziehung zwischen binärer Wertigkeit und Stromkreis Sender-Empfänger (z. B. Computer-Drucker) herzustellen:

0 Es fließt kein Strom
1 Es fließen etwa 20 mA

Bild 2 zeigt eine typische Schnittstelle für diese Technik. Es ist einzusehen, daß es nicht möglich ist, Computer und Drucker mit der gleichen Schaltung für die Schnittstelle auszurüsten, da sonst falsche Pegel auftreten würden. Vielmehr muß z. B. auf der Terminal-Seite ein gleichspannungsfreier Anschluß mit „schwebendem" Bezugspotential vorhanden sein, wie sich das z. B. mit einem Optokoppler, mit einem Reed-Relais oder auch dadurch erreichen läßt, daß der Magnet des Druckers sonst intern keine weiteren Verbindungen als zum Computer aufweist.
Bild 3 gibt daher eine Schaltung wieder, die es gestattet, zwei 20-mA-Schnittstellen miteinander zu verbinden, die beide in der Art von Bild 2 aufgebaut sind. Dazu dienen hier zwei PNP-Transistoren. Es sei erwähnt, daß solche 20-mA-Schnittstellen bei zahlreichen Mikrocomputern anzutreffen sind (KIM-1, AIM-65, PC-100 usw.). Eine Verbindung zweier solcher Computer miteinander läßt sich daher mit Bild 3 einfach realisieren.
Baudot-Fernschreiber arbeiten üblicherweise nicht mit einer 20-mA-Stromschleife, sondern benötigen zur Ansteuerung des Druckers rund 40 mA. Zu beachten ist auch noch, daß der Druckermagnet eine erhebliche Selbstinduktion besitzt, die einen ziemlich „weichen" Stromverlauf auch bei rechteckförmiger Ansteuerung verursacht. Es ist daher empfeh-

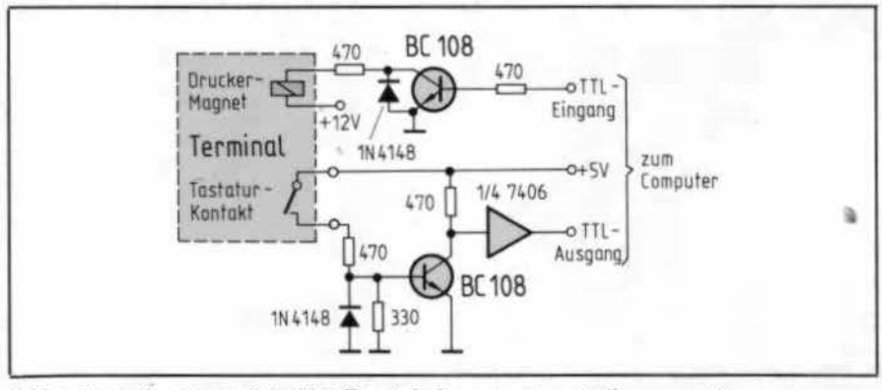


Bild 2. Typische 20-mA-Schnittstellen-Schaltung in einem Mikrocomputer

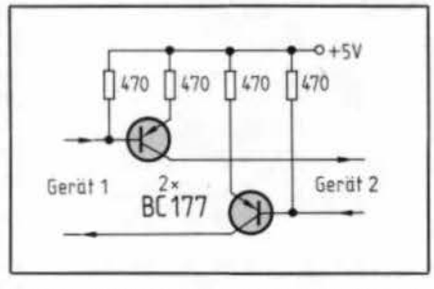


Bild 3. Mit zwei PNP-Transistoren lassen sich zwei der in Bild 2 gezeigten Schnittstellen miteinander verbinden

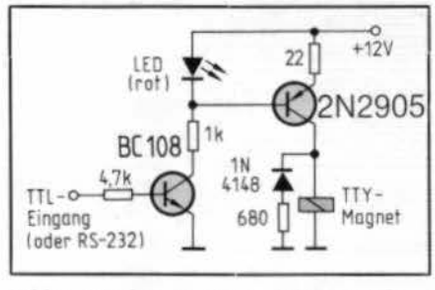


Bild 4. Für die Ansteuerung eines stark induktiven Fernschreiber-Empfangsmagneten empfiehlt sich eine Schaltung, die einen hinreichend steilen Stromanstieg liefert

lenswert, eine besonders dafür konzipierte Schaltung zur Fernschreiber-Ansteuerung einzusetzen, wie sie in *Bild 4* zu sehen ist.

Pegel bei V-24 und RS-232

Eine ebenfalls seriell arbeitende, asynchrone Schnittstelle ist die international genormte V-24-Schnittstelle, die weitgehend der RS-232-Norm entspricht, wie sie von vielen Computer- und Terminal-Herstellern verwendet wird.
Wir wollen uns auf eine vereinfachte Form derselben beschränken, die gewöhnlich für die Verbindung eines Terminals mit dem Computer ausreicht; bei zeilenweise arbeitenden Druckern kann es erforderlich sein, zusätzlich zu den Datenverbindungen noch Steuerleitungen zu verdrahten, um z. B. den Computer zu veranlassen, so lange zu warten, bis der Drucker mit der aktuellen Zeile fertig ist.
Das Datenformat (Startbit, Stopbit(s), Geschwindigkeiten usw.) sieht genauso aus wie bei der 20-mA-Stromschleife, lediglich die Pegel sind etwas anders: Das Null-Potential entspricht dem Pegel −1,5...−36 V, das Eins-Potential +1,5...36 V. Der Bereich von −1,5...+1,5 V ist undefiniert und daher „verboten". Im Gegensatz zur TTY-Schnittstelle sind alle Pegel stets auf eine gemeinsame Masse bezogen, so daß keine Potentialschwierigkeiten auftreten und deshalb gewöhnlich auch keine Optokoppler erforderlich sind. Der „Sender" sollte möglichst niederohmig sein und einige mA Belastung vertragen können; umgekehrt sollte der „Empfänger" einen möglichst hohen Eingangswiderstand besitzen. Diese Forderungen lassen sich z. B. durch die Verwendung von Operationsverstärkern mit symmetrischer Versorgungsspannung realisieren.
Aufgrund der Potentialverhältnisse ist es bei der 20-mA-Stromschleife („TTY" = Teletype) üblich, mit vier Drähten für die beiden Richtungen Terminal-Computer und Computer-Terminal zu arbeiten, während dafür bei der V-24-Norm drei Drähte genügen, weil nur eine gemeinsame Masseleitung nötig ist.
Selbstverständlich ist es ab und zu erforderlich, ein Gerät, das eine 20-mA-Stromschleife besitzt, mit einem solchen zu verbinden, das über eine RS-232-Schnittstelle verfügt. Eine für diesen Zweck geeignete Schaltung ist in *Bild 5* wiedergegeben; um einen 20-mA-Drucker von einer RS-232-Schnittstelle her anzusteuern, läßt sich die in

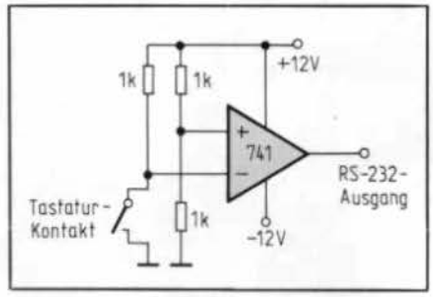


Bild 5. Ein 20-mA-Ausgang eines Terminals läßt sich mit einem preiswerten Operationsverstärker leicht an eine RS-232- oder V-24-Schnittstelle anpassen

Bild 4 angegebene Schaltung einsetzen.

Byte für Byte

Will man Daten oder auch ein Programm von einem Computer zu einem anderen oder auch zu einem EPROM-Programmierer usw. übermitteln, so genügt es natürlich nicht, einfach ein Byte nach dem anderen zu senden. Zusätzlich sind nämlich die folgenden Informationen für den Empfänger notwendig:

1. Wann beginnt die Übertragung gültiger Zeichen? Schließlich muß ja sichergestellt sein, daß nicht irgendwelche Steuerzeichen oder gar Störimpulse vom Empfänger bereits als Daten aufgefaßt werden.
2. Bei der Übermittlung von Maschinenprogrammen ist es nötig, die Anfangsadresse zu übermitteln, damit der empfangende Computer weiß, ab wo er das Programm in seinen Speicher schreiben soll.
3. Bei der Übermittlung von Text ist es erforderlich, ein Zeichen für die Zeilentrennung zu reservieren (üblicherweise CR = hex 0D, optional mit LF = hex 0A).
4. Dem Empfänger muß mitgeteilt werden, wann die Übertragung beendet ist, damit er von Empfang auf lokale Terminalbedienung umschalten kann.
5. Wenn irgend möglich, sollte man dafür sorgen, daß Übertragungsfehler erkannt werden, z. B. durch Verwendung einer Prüfsumme oder eines hinreichend redundanten Codes.

Grundsätzlich sollte ein übertragener Maschinenprogramm-Datenblock also wie folgt aufgebaut sein:

1. Beliebiges, aber eindeutig definiertes Startzeichen.
2. Anfangsadresse der übertragenen Daten.
3. Zeilentrennzeichen oder (alternativ) Angabe der Zeilenlänge.
4. Eindeutig definierter Code, um das Ende der Übertragung zu kennzeichnen.
5. Prüfsumme über alle relevanten Daten.

Selbstverständlich kann die Reihenfolge einiger hier genannter Teildaten vertauscht sein; das tut dem Prinzip keinen Abbruch. Vertauscht man z. B. Punkt 2 und Punkt 3, so gelangt man fast unvermeidlich zu dem von MOS Technology (z. B. KIM-1), Synertek (SYM-1), Rockwell (AIM-65, AIM-65/40 und System-65), Siemens (PC-100) und einigen anderen Firmen verwendeten Lochstreifen-Format, das sich natürlich auch für die Übertragung per Telefon-Modem oder per Funk bestens eignet.

Das MOS-Technology-Format

Wir nennen unser erstes Lochstreifen-Format so, weil es von MOS Technology, der Firma, die den Mikroprozessor 6502 entwickelte, erstmals eingeführt und verwendet wurde.
Für unsere Datenformat-Betrachtungen gehen wir davon aus, daß ab der Adresse 0000 (vierstellig wegen des 64-KByte-Adressenbereichs der meisten heutigen Mikrocomputer) die fünf Bytes 11 22 33 44 55 einzuschreiben sind; dies gilt auch für die später zu besprechenden Formate.
Bei diesem Sachverhalt sähen die übertragenen ASCII-Zeichen wie folgt aus:

```
;05000011223344550104
;0000010001
```

Die Strichpunkte kündigen jeweils den Beginn eines gültigen Datenblockes an. In der ersten Zeile folgt dann die zweistellige hexadezimale Anzahl der Datenbytes, hier 05, anschließend die vierstellige hexadezimale Anfangsadresse 0000. Ohne Zwischenraum oder Trennungszeichen folgen die fünf Datenbytes, gefolgt von einer wiederum vierstelligen hexadezimalen Prüfsumme über alle vorherigen Bytes (inklusive Datenbyte-Anzahl 05 und Startadresse, jedoch ohne den Strichpunkt).
Auf diese erste Zeile könnten nun weitere, gleichartige Datenblöcke folgen, jeweils mit Strichpunkt, Byteanzahl,

Anfangsadresse, Datenbytes und Prüfsumme. Die Zeilen sind untereinander durch eine Wagenrücklauf/Zeilenvorschub-Folge getrennt (CRLF).
Die letzte Zeile enthält keine Datenbytes mehr, deshalb folgt nach dem Strichpunkt auch die Byte-Anzahl 00. Statt dessen wird nun zweimal hintereinander die vierstellige Anzahl der vorher übertragenen Datenblöcke bzw. -zeilen übertragen, hier natürlich 0001. Nach diesen insgesamt immer acht Zeichen nach der Byteanzahl 00 ist die Übertragung beendet. Die zweimalige Nennung der Zeilenzahl ergibt sich automatisch daraus, daß in jeder Zeile eine vierstellige Prüfsumme der vorhergehenden Bytes gesendet wird, die hier identisch mit der Zeilenzahl ist. Das MOS-Technology-Format funktioniert sehr störsicher. Es beinhaltet eine gewisse Redundanz, die auch innerhalb einer Zeile schon die Fehlererkennung ermöglicht, da nur ASCII-Zeichen aus dem Vorrat der hexadezimalen Ziffern vorkommen dürfen.

Das BNPF-Format

Die schon durch die zugelassenen Zeichen gegebene Fehler-Erkennungsmöglichkeit läßt sich natürlich noch verbessern, wenn noch weniger Zeichen erlaubt sind. Ein Extrembeispiel dafür ist das BNPF-Format; Spötter behaupten, es sei von Lochstreifen-Herstellern erfunden worden, die damit ihren Umsatz erhöhen wollen: Es ist nämlich extrem papierverschwendend.
Das BNPF-Format arbeitet nicht mit Hex-Ziffern, sondern überträgt jedes einzelne Bit: Ein N wird für ein Null-Bit und ein P für ein 1-Bit ausgegeben. Zusätzlich wird jedem Byte (bestehend aus acht Bits) ein B voraus- und ein F nachgestellt. Die Übertragung wird mit dem nichtdruckenden Steuerzeichen STX (hex 02 in ASCII) begonnen und mit ETX (hex 03) beendet. Alle Zeichen werden nach der ASCII-Norm gesendet. Die Übertragung unserer fünf Bytes sähe dann so aus:

BNNNPNNNPF
BNNPNNNPNF
BNNPPNNPPF
BNPNNNPNNF
BNPNPNPNPF

Man sieht gleich, daß das genau den Bytes 11 22 33 44 55 entspricht, indem man einfach die Bits mit N oder P einzeln codiert.
Der BNPF-Code ist zwar extrem redundant und arbeitet daher auch ohne Prüfsumme zuverlässig, beinhaltet aber keinerlei Adressennennung. Er ist daher nur für die Übertragung solcher Daten geeignet, die als zusammenhängender Adressenbereich vorliegen und bei denen die Adressenlage dem Empfänger schon vor Beginn der Übertragung bekannt ist (z. B. EPROM-Programmiergeräte). Er stellt daher wohl eines der besten Beispiele dafür dar, wie man mit hohem Datenträger-Aufwand ein Minimum an Information übertragen kann...

Nicht viel besser: B10F und BHLF

Ersetzt man beim BNPF-Format den Buchstaben P durch eine 1 und N durch eine 0, so entsteht das B10F-Format. Da BNPF und B10F sonst völlig identisch miteinander und auch gleich papierfressend sind, wird hier auf den B10F-Code nicht mehr näher eingegangen; er konnte sich ohnehin recht wenig verbreiten.
Ähnliches gilt für den BHLF-Code; bei ihm steht ein H für die 1 und ein L für die 0. Hier sieht man wieder einmal beispielhaft, wie große Mühe sich Computerfirmen geben, ohne jede Argumentation hinsichtlich echten Vorteilen gegenüber Konkurrenten-Codes die totale Inkompatibilität sicherzustellen; sonst könnte ja jemand auf die Idee kommen, zum Computer der Firma X das Peripheriegerät des Konkurrenten Y zu erwerben!

Motorolas Format

Bedeutend intelligenter ist da schon das von Motorola für das Exorciser-Mikrocomputer-System verwendete Datenformat. Jede Zeile der Übertragung beginnt mit dem Buchstaben S. Die erste Zeile enthält einen „File-Namen", der aus ASCII-Zeichen besteht, die aber hexadezimal verschlüsselt sind; ein A erscheint z. B. als 41 (hex). In jeder Datenzeile steht nach dem S zunächst eine Hex-Ziffer für die laufende Zeile (in der ersten Zeile, d. h. jener mit dem Filenamen, steht dafür eine Null). Dann folgt die Byteanzahl in dieser Zeile als zweistellige Hex-Zahl, die Datenbytes und schließlich ein Byte (zwei Hex-Ziffern) als Zeilenprüfsumme. Die letzte übertragene Zeile enthält schließlich keine Datenbytes mehr, sondern nur noch das Adressenfeld und ein Prüfsummen-Byte. Das kann dann so aussehen:

S00800004441544120492F4FF3
S10800001122334455F8
S9030000FC

Das Vorhandensein eines File-Namens ist wohl eine der herausstechenden Eigenschaften dieses Formats.

Das COSMAC-Datenformat

COSMAC heißt ein von RCA entwickeltes Mikrocomputer-System (CPU: 1802). Das dabei verwendete Datenformat ist leider sehr mangelhaft gegen fehlerhafte Übertragung geschützt: Es wird nämlich keine Prüfsumme gesendet. In unserem Beispiel sieht das Format so aus:

!M0000,
1122334455

Das bedeutet, daß ab einer bestimmten Anfangsadresse einfach alle Bytes nacheinander in den Speicher geschrieben werden. Die einzige Fehlererkennungsmöglichkeit ist, daß die einzelnen Zeichen nur dem hexadezimalen Ziffernvorrat entstammen dürfen (0...F). Deshalb ist dieses Format nicht gerade als absolutes Optimum zu betrachten.

ASCII-Hex-Space

Von manchen Herstellern wird ein „ASCII-Hex-Space"-Format favorisiert – obwohl diese Bezeichnung an sich auch für einige der bisher besprochenen Formate zutreffen würde.
„Space" bedeutet, daß die einzelnen Bytes voneinander durch Leerräume getrennt sind. Es gibt Abarten dieses Formats, bei denen diese Leerräume durch Apostrophe (Hochkomma), Komma oder Prozent-Zeichen ersetzt sind. Das ASCII-Hex-Space-Format sieht dann so aus:

$A0000,
11 22 33 44 55
$S00FF,

Vor der ersten Zeile wird außerdem noch ein STX- oder wahlweise ein SOH-Zeichen (ASCII-hex 02 oder 01) als nichtdruckendes Steuerzeichen gesendet. Ferner folgt dem letzten Byte 55 noch das nichtdruckende Steuerzeichen ETX (ASCII hex 03). 00FF ist hier die 16-Bit-Prüfsumme.
Beim ASCII-Hex-Prozent-Format würde die mittlere Zeile
11%22%33%44%55 lauten, der Rest bleibt gleich.

Das Intellec-Format von Intel

Eine gewisse Verbreitung erlangte auch das Intel-Datenformat, wie es für

Entwicklungssysteme der 8080/8085/8086-Prozessorfamilie üblich ist. Jede Zeile beginnt mit einem Doppelpunkt. Die beiden folgenden Hex-Ziffern (ein Byte) geben die Anzahl der in dieser Zeile folgenden Datenbytes an. Dann folgen vier Ziffern für die Adresse des ersten Datenbytes und zwei zur Identifizierung des Record-Typs (hier 00). Nach den fünf Datenbytes unseres Beispiels wird noch ein Prüfsummen-Byte angehängt. Die letzte Zeile beginnt wieder mit einem Doppelpunkt, ist aber durch 01 als Record-Typ gekennzeichnet. Die Byte-Anzahl ist bei ihr 00, und das Adressenfeld ist irrelevant. Unser Beispiel sieht im Intellec-Format so aus:

```
:050000001122334455FC
:00000001
```

Die 8-Bit-Prüfsumme ist im Vergleich zu der z. B. beim Motorola- oder MOS-Technology-Format verwendeten 16-Bit-Prüfsumme nicht ganz so zuverlässig bei der Fehlererkennung, reicht aber normalerweise aus.

Noch'n Format: Tektronix

Qualitativ ist das bei Tektronix-Rechnern verwendete Datenformat dem von Intel durchaus vergleichbar – es ist eben nur anders, um jeder Kompatibilität von vornherein vorzubeugen. Statt dem Doppelpunkt steht zu Beginn jeder Zeile ein Schrägstrich. Es folgen vier Adressenziffern, zwei Ziffern für die Datenbyte-Anzahl, zwei weitere als Zwischen-Prüfsumme von Adresse und Byteanzahl, die fünf Datenbytes und schließlich wieder eine 8-Bit-Prüfsumme in Form zweier Hex-Ziffern. In der letzten Zeile steht wieder ein Adressenfeld (vier Ziffern), die Byteanzahl 00 und zwei Ziffern als 8-Bit-Prüfsumme der letzten Zeile:

```
/00000505112233445 51E
/00000000
```

Die erwähnte Zwischenprüfsumme in jeder Datenzeile entspricht im Format der Prüfsumme der letzten Zeile: Sie steht an der selben Stelle.

Und so weiter und so fort!

Das sind leider noch nicht alle der heute verwendeten Lochstreifen-Datenformate für Objektprogramme; die babylonische Datenverwirrung wird durch mehrere Hex- und Oktalformate noch erweitert, die hier aber nicht mehr genauer erläutert werden. Lediglich Entwickler von EPROM-Programmiergeräten, die mit allen möglichen Computersystemen kompatibel sein müssen, brauchen sich darum zu kümmern [2].
Nun ist natürlich die Frage naheliegend, warum denn jeder Computerhersteller sein Privat-Format erfinden muß. Ist es die Unwissenheit, daß es schon durchaus gut funktionierende Datenformate gibt? Diese Unwissenheit dürfte mit diesem Beitrag vielleicht etwas vermindert worden sein. Oder ist es tatsächlich der bösartige Wille zur Inkompatibilität in Tateinheit mit mißverstandenem Konkurrenzdenken? Fe.

Literatur

[1] *Feichtinger, H.:* Basic für Mikrocomputer. Franzis-Verlag, München.
[2] Data I/O System 19 Manual. Macrotron GmbH, München.

Z-80-Programmier-Tip

Wie ein Unterprogramm merkt, wo man es hingeladen hat

Oft ist es von großem Vorteil, Unterprogramme zu haben, die ganz und gar unabhängig sind von einem bestimmten Speicherbereich, die von allgemeiner Bedeutung sind, wie etwa Ein-/Ausgaberoutinen usw., die man dann beliebig in jedes Programm einbinden kann. Das ist mit relativen Sprüngen beim Z80 leicht möglich. Wenn man dann aber aus diesem Unterprogramm heraus auf eine Monitor-Subroutine springen will, bei der man z. B. bei Textausgaben eine absolute Adresse übergeben muß, die Bestandteil des relokatiblen Unterprogramms ist, stößt man auf Schwierigkeiten. Woher weiß man beim Schreiben der Routine, auf welche Adresse der Datenbereich, den man ja mitverschieben will, später zu liegen kommt.

Eine Möglichkeit ist, zurückzuschauen, unter welcher Adresse das Unterprogramm aufgerufen wurde. Die Aufrufstelle hat man durch die Return-Adresse im Stack. Den Rest kann man durch die relative Distanz des Datenbereichs zur Anfangsadresse berechnen. Man muß lediglich die Reihenfolge im Stack etwas umordnen und am Schluß wieder rückwärts abspulen – so wird kein Register dabei zerstört. Alles weitere kann man aus dem Beispiel ersehen. Hier wurde die absolute Adresse im Register DE an die Monitor-Subroutine „MSG" übergeben. Sehr gut geeignet wäre auch das IX- oder IY-Register, da hier indizierte Adressierung möglich ist. *Karl Schedler*

Programmbeispiel

„Rufendes" Hauptprogramm:

```
        .....
        CALL RELOC ;Aufruf
RETAD:  .....
        .....
```

Relokatibles Unterprogramm, eingebunden ab Adresse RELOC

```
RELOC:  EX (SP), HL     ;hole
                        Return-Adr.
                        rette HL
        PUSH HL         ;rette
                        Return-Adr.
        PUSH DE         ;rette DE
        .....
        DEC HL          ;MSB
                        Aufruf-Adr.
        LD D,(HL)
        DEC HL          ;LSB
                        Aufruf-Adr.
        LD E,(HL)       ;RELOC
                        in DE
        .....
        LD HL,MES-RELOC
        ADD HL,DE       ;berechne
        .....           MES
        EX DE,HL
        CALL MSG        ;Aufruf
                        MSG übergeben
        .....           MES in DE

MES:    DEFW            ;Textzeichen
        .....
ENDE:   POP DE          ;hole DE
        POP HL          ;hole
                        Return-Adr.
        EX (SP),HL      ;hole HL,
                        Push Return-
RET                     Adr. RETAD
```

Jürgen Plate

Suchen und Sortieren in Pascal und Basic

1. Teil

In der folgenden Serie sollen die wichtigsten Verfahren zum Sortieren von Daten und zum Suchen nach einem bestimmten Element einer Datenmenge vorgestellt werden. Die Anwendungsbereiche dafür lassen in ihrer Vielzahl nichts zu wünschen übrig. Gleichzeitig sind es aber auch die Sortier- und Suchalgorithmen, die die meiste Rechenzeit vieler Programme verschlingen, und so ist gerade hier die Notwendigkeit am größten, einen effizienten Algorithmus zu finden. Alle hier vorgestellten Algorithmen sind getestete Unterprogramme, abgefaßt in den beiden Programmiersprachen Pascal und Basic. Die Verbreitung dieser Sprachen im Mini- und Mikrocomputerbereich läßt fast immer eine direkte Verwendung der Unterprogramme zu. In allen Programmen wurde auf Klarheit und Übersichtlichkeit geachtet, so daß eine Adaption an einen bestimmten „Dialekt" oder die Übertragung in eine andere Programmiersprache keine Schwierigkeiten bereiten dürfte.

Tabelle 1: Anweisungsstrukturen

Pascal	Basic

IF <Bedingung> THEN
  <Anweisung 1>
ELSE
  <Anweisung 2>;
	100 IF <Bedingung> THEN 300
	200 <Anweisung 2>
	250 GOTO 400
	300 <Anweisung 1>
	400 REM Hier geht es weiter

WHILE <Bedingung> DO <Anweisung>;	100 IF NOT <Bedingung> THEN 400 200 <Anweisung> 300 GOTO 100 400 REM Hier geht es weiter
REPEAT <Anweisung(en)> UNTIL <Bedingung>;	100 <Anweisung(en)> 200 IF NOT <Bedingung> THEN 100
FOR <Variable>: = <Wert> TO <Endwert> DO <Anweisung>;	100 FOR <Variable> = <Wert> TO <Endwert> 200 <Anweisung> 300 NEXT <Variable>

<Wert> und <Endwert> können Ausdrücke sein.

Die wichtigsten Anweisungsstrukturen von Pascal und Basic

Die Erläuterung der Algorithmen erfolgt an den Pascal-Unterprogrammen, da bei dieser Sprache die Daten- und Anweisungsstrukturen wesentlich übersichtlicher sind. Für diejenigen unter Ihnen, denen Pascal nicht so geläufig ist, sind die wichtigsten Anweisungsstrukturen der beiden Sprachen in *Tabelle 1* einander gegenübergestellt.

Sollen in Pascal hinter den Schlüsselworten THEN, ELSE oder DO mehrere Anweisungen stehen, so sind diese mit den Schlüsselworten BEGIN und END zu „klammern". Sie sehen das in den Programmen. Die Schlüsselworte zur Datendefinition und -Strukturierung (CONST, TYPE, VAR) brauchen Sie als Basic-Programmierer nicht; die Erklärungen zu den Programmen verwenden sie zwar, sind aber auch für „Nur-Basic-Programmierer" gut zu verstehen.

Alle Pascal-Unterprogramme werden mit dem Schlüsselwort PROCEDURE eingeleitet und die „von außen" kommenden Variablen als Parameter mitgegeben. So ist zum Beispiel bei der

```
PROCEDURE SORT
(VAR D : DATA);
```

zum Sortieren der Kundendaten der Aufruf

```
SORT(KUNDEN);
```

irgendwo im Hauptprogramm möglich.

Haben die Lieferanten den gleichen Aufbau (die gleiche Struktur) wie die

Kunden, können auch diese mit

```
SORT(LIEFERANTEN);
```

sortiert werden. Dieser Prozedurmechanismus hat einen weiteren Vorteil: Alle innerhalb der Prozedur definierten Variablen haben nur innerhalb der Prozedur Gültigkeit und sind nur dort vorhanden. Gleichnamige Variablen des Hauptprogramms werden nicht beeinflußt. In Basic ist ein derartiger Mechanismus nicht verfügbar (leider!).

Datenstrukturen und ihre Realisierung in den beiden Sprachen

Grundsätzlich wird die zu sortierende Information unterteilt in das Such- oder Sortierkriterium (Schlüssel, Key) und die eigentliche Information. Zum Beispiel ist bei einem Datensatz der Kundendatei der Schlüssel die Kundennummer. Die Information besteht aus Namen, Adresse, Bankverbindung, offenen Rechnungen usw.

Es ist aber in manchen Fällen möglich, daß der Schlüssel gleichzeitig die einzige Information darstellt (z. B. Namenslisten). In den Programmen wird als Schlüssel eine Integerzahl (Ganzzahl) verwendet; sie läßt sich natürlich durch jede andere Datenstruktur ersetzen, die direkte Vergleiche zuläßt. Als Information wird in den Programmen eine Zeichenkette (String) verwendet. In Pascal sieht das dann so aus wie in *Tabelle 2*.

Bei der Variablen des Types ITEM werden durch die Wertzuweisung V1 := V2 alle Teile des Records (also KEY und INFO) von V1 nach V2 gebracht. Die einzelnen Komponenten von V1 lassen sich durch den Selektionsmechanismus V1.KEY bzw. V1.INFO ansprechen.

In Basic gibt es keine derartige Datenstrukturierungsmöglichkeit. Es werden hier als Behelf zwei getrennte Variablen verwendet:

K für den Schlüssel (Key)
I$ für die Information (Info)

Ist der Schlüssel gleichzeitig die einzige Information, vereinfacht sich die Struktur natürlich erheblich. Die einzelnen Datensätze werden in linearen Feldern (Arrays) gespeichert. So ergeben sich die Programme in *Tabelle 3*.

Tabelle 2: Datentyp-Definition in Pascal

```
TYPE ITEM =  RECORD
        KEY :  INTEGER;
       INFO :  PACKED ARRAY [1..10] OF CHAR
               END;
```

Tabelle 3: Datentypen-Definition im Programm

Pascal	Basic
CONST N = 1000; (* Feldlänge *)	10 N = 1000
TYPE STRING = PACKED ARRAY	20 REM N ist als konstant
[1...10] OF CHAR;	30 REM zu betrachten
TYPE ITEM = RECORD	40 DIM K(1000), I$(1000)
KEY : INTEGER;	
INFO : STRING	
END;	
DATA : ARRAY [1..N] OF ITEM;	
VAR D : DATA;	

```
PROCEDURE LINSORT (VAR D : DATA);
   (*  SORTIEREN DES FELDES D NACH DEN SCHLUESSELN D[..].KEY
       IN AUFSTEIGENDER FOLGE.
       GLOBALE KONSTANTE : N  FELDLAENGE
       GLOBALE TYPEN: ITEM = RECORD  KEY : ....; INFO : .... END;
                      DATA = ARRAY [1..N] OF ITEM;
   *)
   VAR I,J :0..N;
       H : ITEM;
       ENDE : BOOLEAN;
   BEGIN
   FOR I := 2 TO N DO
      BEGIN
      ENDE := FALSE;
      H := D[I];
      J := I - 1;
      WHILE NOT ENDE AND (J>0) DO
         IF H.KEY < D[J].KEY THEN (* WEITERSUCHEN *)
            BEGIN D[J+1] := D[J]; J := J - 1 END
         ELSE ENDE := TRUE;
      D[J+1] := H;
      END;
   END (* LINSORT *);
```

Bild 1. Pascal-Programm für lineares Sortieren eines Datenfeldes

```
10000 REM * * * * * * * * * * LINSORT * * * * * * * * * *
10001 REM * SORTIEREN DES FELDES I$ NACH DEN SCHLUESSELN
10002 REM * IN DEM FELD K.
10003 REM * DIE ELEMENTE K(J) UND I$(J) BILDEN EIN PAAR
10004 REM * SCHLUESSEL - INFORMATION.
10005 REM * DIE UNTERE FELDGRENZE IST 1, DIE OBERE FELD-
10006 REM * GRENZE STEHT IN DER VARIABLEN  N.
10010 FOR I =  2 TO N
10020   H$ =I$(I)
10030   H = K(I)
10040   J=I-1
10050 IF H>=K(J) THEN 10100
10060   K(J+1)=K(J)
10070   I$(J+1)=I$(J)
10080   J = J - 1
10090   IF J<>0 THEN 10050
10100   K(J+1) = H
10110   I$(J+1) = H$
10120 NEXT I
10130 RETURN
10140 REM * * * * * * * * * * * * * * * * * *
```

Bild 2. Basic-Unterprogramm für lineares Sortieren von Feldvariablen

```
PROCEDURE BINARYSORT (VAR D : DATA);
   (*  SORTIEREN DES FELDES D NACH DEN SCHLUESSELN D[..].KEY
       IN AUFSTEIGENDER FOLGE.
       GLOBALE KONSTANTE : N  FELDLAENGE
       GLOBALE TYPEN: ITEM = RECORD  KEY : ....; INFO : .... END;
                      DATA = ARRAY [1..N] OF ITEM;
   *)
   VAR I,J,L,R,M : 0..N;
       H : ITEM;
   BEGIN
   FOR I := 2 TO N DO
   BEGIN
      J := I - 1;
      L := 1; R:=J;
      H := D[I];
      WHILE L <= R DO (* BINAERE SUCHE *)
         BEGIN
         M := (L + R) DIV 2;
         IF H.KEY < D[M].KEY THEN R := M - 1 ELSE L := M + 1;
         END;
      WHILE J >= L DO (* VERSCHIEBEN *)
         BEGIN D[J+1] := D[J]; J :=J - 1 END;
      D[L] := H;
      END;
   END (* BINARYSORT *);
```

Bild 3. Sortieren durch binäres Einfügen in Pascal

```
20000 REM * * * * * * * * * * BINARY SORT * * * * * * * * * *
20001 REM * SORTIEREN DES FELDES I$ NACH DEN SCHLUESSELN
20002 REM * IN DEM FELD K.
20003 REM * DIE ELEMENTE K(J) UND I$(J) BILDEN EIN PAAR
20004 REM * SCHLUESSEL - INFORMATION.
20005 REM * DIE UNTERE FELDGRENZE IST 1, DIE OBERE FELD-
20006 REM * GRENZE STEHT IN DER VARIABLEN  N.
20010 FOR I = 2 TO N
20020    L=1
20030    R= I - 1
20040    H$ = I$(I)
20050    H = K(I)
20060    REM * * * BINAERE SUCHE * * *
20070    IF L>R THEN 20140
20080    M= INT( (L+R)/2 )
20090    IF H<K(M) THEN 20120
20100    L=M+1
20110    GOTO 20070
20120    R = M-1
20130    GOTO 20070
20140    FOR J = I-1 TO L STEP -1
20150       I$(J+1) = I$(J)
20160       K(J+1) = K(J)
20170    NEXT J
20180    I$(L) = H$
20190    K(L) = H
20200 NEXT I
20210 RETURN
20220 REM * * * * * * * * * * * * * * * * * *
```

Bild 4. Basic-Programm zum Sortieren durch binäres Einfügen

```
PROCEDURE BUBBLESORT(VAR D : DATA);
   (*  SORTIEREN DES FELDES D NACH DEN SCHLUESSELN D[..].KEY
       IN AUFSTEIGENDER FOLGE.
       GLOBALE KONSTANTE : N  FELDLAENGE
       GLOBALE TYPEN: ITEM = RECORD  KEY : ....; INFO : .... END;
                      DATA = ARRAY [1..N] OF ITEM;
   *)
   VAR J,I : 0..N;
   PROCEDURE SWAP(VAR X,Y : ITEM);
      VAR Z : ITEM;
      BEGIN Z := X; X := Y; Y := Z END;
   BEGIN
   FOR I := 2 TO N DO
      FOR J := N DOWNTO I DO
         IF D[J-1].KEY > D[J].KEY THEN SWAP(D[J],D[J-1]);
   END (* BUBBLESORT *);
```

Bild 5. Pascal-Prozedur für „Bubblesort" (Blasen-Sortieren)

Programmerstellung, Effizienzvergleich

Es wurde versucht, die Programme so übersichtlich wie möglich und möglichst universell verwendbar zu machen. So wurde in Basic auf die Besonderheiten bestimmter Dialekte verzichtet. In den Pascal-Listings wurden die Schlüsselworte (BEGIN, END usw.) aus Gründen der Übersichtlichkeit fett gedruckt. Ebenso sind in den Programmtext erläuternde Kommentare (*...*) eingefügt.

Der Aufruf der Pascal-Prozeduren erfolgt wie üblich durch den Namen, der Aufruf der Basic-Unterprogramme geschieht einheitlich mit GOSUB. Für jedes Pascal-Unterprogramm wurde die Effizienz in Form einer Zeitmessung geprüft: Die Rechenzeit wurde einmal für die Bearbeitung von 100 und einmal für die Bearbeitung von 1000 Feldelementen durchgeführt. Dabei wurde jeder Datensatz einmal sortiert, einmal umgekehrt sortiert und einmal rein zufällig angeordnet von dem Programm bearbeitet.

Da die Basic-Programme aus den Pascal-Algorithmen entwickelt wurden, lassen sich die Ergebnisse des Effizienzvergleichs der einzelnen Sortierverfahren im Groben auch auf die Basic-Programme übertragen.

Sortieren

Jetzt kann's losgehen – wir haben Datentypen definiert und somit die Grundlage geschaffen, mit diesen Daten irgend etwas anzufangen. Wenden wir uns zunächst dem Sortieren von Feldern zu. Später werden wir einen Effizienzvergleich der besprochenen Verfahren anstellen.

Lineares Einfügen

Dieses einfachste aller Sortierverfahren nimmt beginnend bei I=2 das I-te Datenelement und fügt es an der entsprechenden Stelle im Datenfeld ein. Danach wird I um 1 erhöht und solange fortgefahren, bis alle Datenelemente bearbeitet sind. Sind nur wenige Daten zu sortieren, hat auch dies einfache Verfahren durchaus seine Berechtigung. Der Algorithmus lautet also:

FOR I := 2 TO N DO
  (* füge den Wert von D[I] an entsprechender Stelle in D[1]...D[I] ein *)

Das Einfügen geschieht, indem für J von I-1 ausgehend das Element D[I] mit dem Element D[J] verglichen wird und dann entweder D[I] eingefügt oder D[J] nach rechts verschoben wird. Also muß das Element D[I] vorher auf einer Hilfsvariablen gespeichert werden. *Bild 1* zeigt die Pascal-Prozedur und *Bild 2* das Basic-Unterprogramm.

Binäres Einfügen

Der vorhergehende Algorithmus läßt sich verbessern, indem die bereits vorhandene Ordnung der Elemente D[1] bis D[I] ausgenützt wird, um die Einfügungsstelle zu finden. Beim binären Einfügen wird in der Mitte der Elemente D[1] bis D[I] geprüft, ob H in der unteren oder in der oberen Hälfte eingefügt werden muß. Die entsprechende Teilfolge wird wieder halbiert, bis durch die Teilung die Einfügungsstelle gefunden worden ist.

Leider ergibt sich nur eine Reduzierung in der Zahl der Vergleiche, denn der Rest des Programms, also auch die Zahl der Verschiebungen, ist der gleiche wie bei der linearen Sortierung. *Bild 3* enthält das Pascal-Listing, *Bild 4* das Basic-Listing.

Bubblesort und Shakersort

Bei diesen beiden Verfahren wird wie bei den vorhergehenden das Feld mehrfach durchlaufen. Anstelle des Einfügens des entsprechenden Feldelements an der richtigen Stelle wird hier das jeweils kleinste Element durch Austausch mit dem Nachbarn an die passende Stelle gebracht. Wenn man das Feld vertikal aufschreibt, „perlt" das Feldelement nach oben an seinen Platz, und das hat diesem Verfahren den Namen eingetragen. In der Pascal-Version wird der Austausch durch die Prozedur SWAP erledigt, in der Basic-Version sind die Operationen in das Programm eingefügt (*Bild 5* und *Bild 6*).

Wenn Sie den Algorithmus auf die Möglichkeit der Verbesserung hin untersuchen, fällt Ihnen sicher gleich die feste Schleifenstruktur auf, die keine Rücksicht auf schon sortierte Sequenzen nimmt. Die erste Verbesserung ist dadurch möglich, wenn das Programm abbricht, wenn das Feld sortiert ist. Das ist der Fall, wenn in einem Durchlauf für i kein Austausch erfolgte.

Weiterhin ist Bubblesort höchst asymmetrisch, wenn Sie zum Beispiel die beiden folgenden Felder betrachten, können Sie feststellen, daß D1 schon nach einem Durchlauf, D2 dagegen erst nach sieben Durchläufen sortiert ist.

D1:	10	25	36	54	87	2
D2:	87	2	10	25	36	54

Die Asymmetrie läßt sich beseitigen, indem in aufeinanderfolgenden Durchläufen jeweils in entgegengesetzter Richtung gearbeitet wird. Dieses „Auf" und „Ab" hat dem Verfahren auch zu dem Namen Shakersort verholfen. Eine letzte Verbesserung ergibt sich, wenn man den Index festhält, bis zu dem das Feld schon sortiert ist. Das Ergebnis der Verbesserung von Bubblesort finden Sie in *Bild 7* und *Bild 8*.

Im übrigen sind von Bubblesort und Shakersort keine besonders große Verbesserungen zu erwarten. Shakersort ist jedoch bei weitgehend sortierten Feldern zu empfehlen.

Mit diesen beiden Verfahren sind die einfachen Sortiermethoden abgeschlossen. Die beiden folgenden sehr effizienten Sortierverfahren sind in ihrem Algorithmus wesentlich komplizierter und finden hauptsächlich dann Anwendung, wenn es darum geht, sehr große Datenmengen zu sortieren.

```
30000 REM * * * * * * * * * * BUBBLESORT * * * * * * * * * * *
30001 REM * SORTIEREN DES FELDES I$ NACH DEN SCHLUESSELN
30002 REM * IN DEM FELD K.
30003 REM * DIE ELEMENTE K(J) UND I$(J) BILDEN EIN PAAR
30004 REM * SCHLUESSEL - INFORMATION.
30005 REM * DIE UNTERE FELDGRENZE IST 1, DIE OBERE FELD-
30006 REM * GRENZE STEHT IN DER VARIABLEN  N.
30010 FOR I = 2 TO N
30020    FOR J = N TO I STEP -1
30030       IF K(J-1)>K(J) THEN 30050
30040       GOTO 30110
30050       H$ = I$(J)
30060       H = K(J)
30070       I$(J) =I$(J-1)
30080       K(J) = K(J-1)
30090       I$(J-1) = H$
30100       K(J-1) = H
30110    NEXT J
30120 NEXT I
30130 RETURN
30140 REM * * * * * * * * * * * * * * * * * *
```

Bild 6. Bubblesort auf einem Basic-Computer

```
PROCEDURE SHAKERSORT (VAR D : DATA);
   (*  SORTIEREN DES FELDES D NACH DEN SCHLUESSELN D[..].KEY
       IN AUFSTEIGENDER FOLGE.
       GLOBALE KONSTANTE : N  FELDLAENGE
       GLOBALE TYPEN: ITEM = RECORD  KEY : ....; INFO : .... END;
                      DATA = ARRAY [1..N] OF ITEM;
   *)
   VAR I,J,R,L : 0..N;
   PROCEDURE SWAP(VAR X,Y : ITEM);
      VAR Z : ITEM;
      BEGIN Z := X; X := Y; Y := Z END;
   BEGIN
   L := 2;  R := N;  I := N - 1;
   REPEAT
      FOR J := R DOWNTO L DO (* RAUFSCHUETTELN *)
         BEGIN
         IF D[J-1].KEY > D[J].KEY THEN
           BEGIN SWAP(D[J],D[J-1]); I := J (* INDEX MERKEN *) END;
         END;
      L := I + 1;
      FOR J := L TO R DO (* RUNTERSCHUETTELN *)
         BEGIN
         IF D[J-1].KEY > D[J].KEY THEN
           BEGIN SWAP(D[J],D[J-1]); I :=J (* INDEX MERKEN *) END;
         END;
      R := I - 1;
   UNTIL L > R;
   END;
```

Bild 7. Pascal-Prozedur für „Shakersort"

Tabelle 4: Pascal-Programm HEAPSORT

```
PROCEDURE HEAPSORT (VAR D : DATA);
    .
    .
    .
BEGIN
L := (N DIV 2) + 1; (* BINAERER BAUM! *)
R := N;
REPEAT
    IF L1> THEN
       (* HEAP AUFBAUEN *)
       L := L – 1
    ELSE
        IF R1> THEN
           (* HEAP ABARBEITEN *)
           BEGIN SWAP (D[L], D[R]); R := R –1 END;
        (* NAECHSTES HEAPELEMENT DURCH DEN HEAP
        WANDERN LASSEN, SIEHE OBEN *)
UNTIL R =1
END;
```

```
40000 REM * * * * * * * * * * SHAKERSORT * * * * * * * * * * *
40001 REM * SORTIEREN DES FELDES I$ NACH DEN SCHLUESSELN
40002 REM * IN DEM FELD K.
40003 REM * DIE ELEMENTE K(J) UND I$(J) BILDEN EIN PAAR
40004 REM * SCHLUESSEL - INFORMATION.
40005 REM * DIE UNTERE FELDGRENZE IST 1, DIE OBERE FELD-
40006 REM * GRENZE STEHT IN DER VARIABLEN  N.
40010 L = 2
40020 R = N
40030 I = N - 1
40040 REM BEGINN DER REPEAT- SCHLEIFE, ENDE BEI 40170
40050 FOR J = R TO L STEP -1
40060    IF K(J-1) <= K(J) THEN 40090
40070    GOSUB 40190
40075    REM GO AND SWAP D(J) WITH D(J-1)
40080    I = J
40090 NEXT J
40100 L = I + 1
40110 FOR J = L TO R
40120    IF K(J-1)<=K(J) THEN 40150
40130    GOSUB 40190
40140    I = J
40150 NEXT J
40160 R = I - 1
40170 IF L <= R THEN 40040
40180 RETURN
40190 REM * * SWAP (D(I),D(J-1))
40200 H$ = I$(J)
40210 H = K(J)
40220 I$(J) = I$(J-1)
40230 K(J) = K(J-1)
40240 I$(J-1) = H$
40250 K(J-1) = H
40260 RETURN
40270 REM * * * * * * * * * * * * * * * * * * *
```

Bild 8. Basic-Unterprogramm für das Shaker-Sortierverfahren

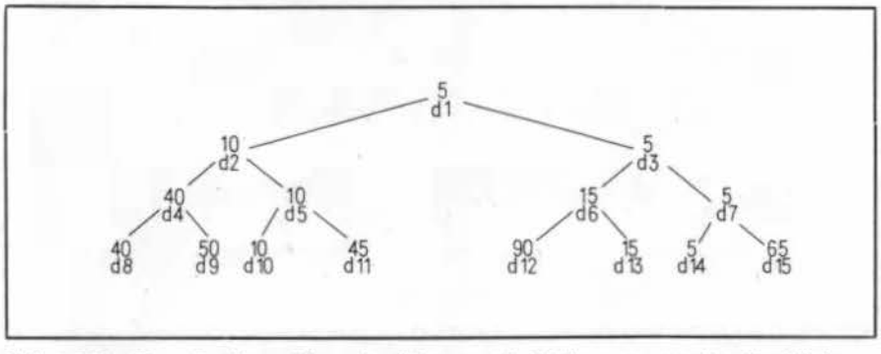


Bild 9. Beim Heap-Sortierverfahren benötigt man für N Elemente stets N-1 Vergleiche

Heapsort

Die bisher behandelten Algorithmen suchten immer nur ein Element, das kleinste aus der Restmenge von N-1 Elementen heraus. Die Sortiermethode läßt sich also nur beschleunigen, indem bei einem Durchlauf mehr Information gesammelt wird. Eine Sortiermethode, die nach diesem Schema verfährt, ist das Sortieren mit Bäumen. Dabei wird als Baum die unten gezeigte Anordnung der Elemente einer Datenmenge bezeichnet. Betrachten Sie dazu die Folge von Schlüsseln:

40 50 10 45 90 15 5 65

Ordnet man diese Elemente in einem Baum derart, daß immer das kleinere Element als Wurzel verwendet wird, benötigt man N-1 Vergleiche (*Bild 9*). Wenn ein solcher binärer (= zweiästiger) Baum in einem linearen Feld gespeichert werden soll, ergibt sich die Bedingung:

d [i] <= d [2i] und d [i] <= d [2i+1]

(Die Feldkomponenten stehen unter den Schlüsseln.) Eine derartige Anordnung der Schlüssel heißt HEAP. Angenommen, der Heap (*Bild 10*) soll um ein Element (d1 = 40) erweitert werden.

Das neue Element (z. B.: d1 = 40) wird zuerst einmal an die Spitze gesetzt. Dann läßt man es entlang der kleineren Elemente nach unten wandern, wobei die kleineren Elemente gleichzeitig nach oben steigen (*Bild 11*).

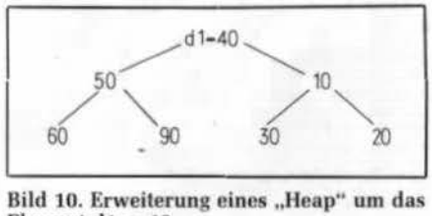


Bild 10. Erweiterung eines „Heap" um das Element d1 = 40

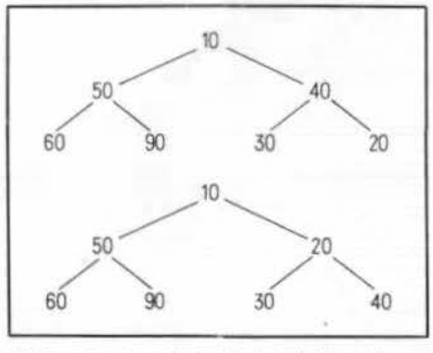


Bild 11. Sortierschritte beim Einfügen des neuen Elements 40

```
PROCEDURE HEAPSORT (VAR D : DATA);
   (*  SORTIEREN DES FELDES D NACH DEN SCHLUESSELN D[..].KEY
       IN AUFSTEIGENDER FOLGE.
       GLOBALE KONSTANTE : N  FELDLAENGE
       GLOBALE TYPEN: ITEM = RECORD  KEY : ....; INFO : .... END;
                      DATA = ARRAY [1..N] OF ITEM;
   *)
   VAR I,J,R,L : INTEGER;
       H : ITEM;
       CONTINUE : BOOLEAN (* STOPPER FUER DIE WANDERUNG *);
   PROCEDURE SWAP(VAR X,Y : ITEM);
      VAR Z : ITEM;
      BEGIN Z := X; X := Y; Y := Z END;
   BEGIN
   L := (N DIV 2) + 1;  R := N;
   REPEAT
      IF L > 1 THEN
        L := L - 1
      ELSE
         IF R > 1 THEN
            BEGIN SWAP(D[L],D[R]); R := R - 1; END;
      (* WANDERN DES NAECHSTEN ELEMENTS DURCH DEN HEAP *)
      I := L;  J := 2*I;
      H := D[I];
      CONTINUE := J <= R;
      WHILE CONTINUE DO
         BEGIN
         IF J < R THEN (* DANN DARF D[J] MIT D[J+1] VERGLICHEN WERDEN *)
            IF D[J].KEY < D[J+1].KEY THEN J := J + 1;
         IF J<=R THEN  (* VERGLEICH H<D[J].KEY ZULAESSIG *)
            CONTINUE := H.KEY < D[J].KEY ELSE CONTINUE := FALSE;
         IF CONTINUE THEN
            BEGIN (* EINORDNEN *)
            D[I] := D[J];
            I := J;  J := 2*I ;
            END;
         END (* WHILE CONTINUE *);
      D[I] := H;
   UNTIL R = 1;
   END (* HEAPSORT *);
```

Bild 12. Die fertige Heapsort-Prozedur in Pascal

Das Sortierverfahren von Heapsort (*Tabelle 4*) besteht also aus zwei getrennten Tätigkeiten:

– Aufbau des Heap aus den Feldelementen (im Programm: IF L<1);
– Abarbeiten des Heap (im Programm: ELSE).

Die fertige Prozedur ist in *Bild 12*, das entsprechende Basic-Programm in *Bild 13* aufgelistet. Diese Prozedur hat noch eine schöne Eigenschaft. Wenn man bei den zwei Zeilen

```
IF D[J].KEY<D[J+1].KEY THEN
J := J + 1;
CONTINUE := H.KEY<D[J].KEY ELSE
CONTINUE := FALSE;
```

die Kleinerzeichen durch Größerzeichen ersetzt werden, wandert das größere Element nach oben und das Feld wird dann absteigend, also genau umgekehrt sortiert; im Basic-Programm sind das die Zeilen 50205 und 50225. Als letztes Sortierverfahren folgt im nächsten Heft Quicksort.

(Fortsetzung folgt)

```
50000 REM * * * * * * * * * * HEAPSORT * * * * * * * * * *
50001 REM * SORTIEREN DES FELDES I$ NACH DEN SCHLUESSELN
50002 REM * IN DEM FELD K.
50003 REM * DIE ELEMENTE K(J) UND I$(J) BILDEN EIN PAAR
50004 REM * SCHLUESSEL - INFORMATION.
50005 REM * DIE UNTERE FELDGRENZE IST 1, DIE OBERE FELD-
50006 REM * GRENZE STEHT IN DER VARIABLEN  N.
50010 L = INT(N/2) + 1
50015 REM IN PASCAL L = N DIV 2+ 1
50020 R = N
50030 IF L>1 THEN 50130
50035 REM BAUM AUFBAUEN
50040 IF R<=1 THEN 50120
50050 H$=I$(L)
50060 H = K(L)
50070 I$(L)=I$(R)
50080 K(L)=K(R)
50090 I$(R)=H$
50100 K(R)=H
50110 R=R - 1
50120 GOTO 50140
50130 L=L-1
50140 REM  W A N D E R N
50150 I=L
50160 J=2*I
50170 H=K(I)
50180 H$=I$(I)
50190 IF J>R THEN 50280
50200 IF J>=R THEN 50220
50205 IF K(J)>=K(J+1) THEN 50220
50210 J = J + 1
50220 IF J>R THEN 50280
50225 IF H>=K(J) THEN 50280
50230 I$(I)=I$(J)
50240 K(I)=K(J)
50250 I=J
50260 J=2*I
50270 GOTO 50200
50280 I$(I)=H$
50290 K(I)=H
50300 IF R<> 1 THEN 50030
50310 RETURN
50320 REM * * * * * * * * * * * * * * * * * *
```

Bild 13. Heapsort läßt sich auch in Basic relativ einfach programmieren

Computer + Elektronik-Rechner

TEXAS-INSTRUMENTS

TI-35 C	52.–
TI-30 LCD	37.–
TI-44	74.–
TI-51 III	85.–
TI-53	55.–
TI-57	73.–
TI-58 C	209.–
TI-59	374.–
PC 100 C	439.–
TI-Programmer	147.–
TI-99/4 Home Computer	1674.–
Monitor BGC 370	978.–

MODULE für TI 58/59

Statistik	120.–
Mathe	143.–
E-Technik	143.–
Spiele	143.–
Baustatik I	259.–
Baustatik II	324.–

HEWLETT-PACKARD

HP 32 E	124.–
HP 33 C	231.–
HP 34 C	339.–
HP 67	922.–
HP 97	1890.–
HP 41 C	593.–
HP 41 CV	763.–
dazu Printer	899.–
Kartenleser	499.–
Memory-Modul	79.–
Opt. Lesestift	299.–
HP 83	auf Anfrage
HP 85	auf Anfrage

APPLE (Euro-Apple II plus)

16 KB	2650.–
32 KB	2750.–
48 KB	2825.–
Disk. m. Controller	1645.–
UHF-Modulator	58.–

EPSON-Drucker für APPLE u. CBM

MX80F/T	1798.–
CBM-Interface	248.–
Apple-Interface	298.–

COMMODORE

CBM 4016-2/3	2590.–
CBM 4032	2990.–
CBM 8032	3790.–
CBM 4040	2984.–
CBM 8050	3790.–
CBM C 2 N	258.–
HEW 1001 (ähnlich CBM 8026)	3498.–
CBM 4001-16 KB	2339.–
CBM 4001-32 KB von HEW erweitert	2583.–

Alle Preise inkl. MwSt. zuzüglich Versandkosten. Vollständige serienmäßige Ausrüstung, Preisänderungen vorbehalten. **Ausführliche Unterlagen gegen Schutzgebühr DM 2.–.** Lieferung an Privatperson nur gegen Nachnahme bzw. Voreinsendung eines Verrechnungs-Schecks, sonst auch gegen Rechnung. Volle Garantie.

HEW-Computer-Technik

Zum Wiesengrund 27, Postfach 3188, 5810 Witten 3
Telefon (0 23 02) 7 32 31/7 32 47, Telex 8 229 164

Beratung
Vertrieb
Service

Walter Karsten

Der Strichcode

Auf jeder Milchtüte, jedem Gurkenglas und jeder Kaffeepackung finden wir heute seltsame Markierungen aus schmalen und breiten Balken: Strichcode, Balkencode oder auch Barcode genannt. Was hat es damit auf sich? Wie setzen sich die einzelnen Zeichen zusammen, und mit welchen Mitteln kann man sie entziffern? – Fragen, die der folgende Beitrag beantwortet.

Die Entwicklung der letzten Jahre zeigte eine nicht übersehbare Tendenz zur dezentralen Datenverarbeitung. Ohne hier auf Pro und Kontra der einzelnen Philosophien einzugehen, läßt sich der zwangsweise gestiegene Bedarf an externer Datenspeicherung nicht übersehen, genauso wie bei der mobilen Datenerfassung. Zu den bisher bekannten Medien gehören nichtflüchtige Halbleiterspeicher, Magnetband, Magnetplatte, Kassette und vielerlei Varianten. Die wesentlichen Unterschiede entstanden wie üblich durch Hausnormen der größten Hersteller, die in bekannter Form ihre Vorstellungen durchsetzten.

Datenträger im Vergleich

Eine Sonderstellung nehmen seit jeher Datenträger ein, die eine optisch erkennbare Information enthalten und maschinell gelesen werden können. Ursprünglich begann das mit den bekannten 80spaltigen Lochkarten, die schon auf Tabuliermaschinen gelaufen waren. Ein altgedienter DV-Praktiker, wohlgemerkt noch nicht EDV, konnte ohne Anlagenhilfe „frei Auge" die Stanzung übersetzen. Das gleiche galt sinngemäß für den Lochstreifen, der – obwohl schon häufig totgesagt – immer noch fröhliche Urständ feiert.
Die durchaus logische Fortentwicklung führte zu einer Form von externer Datenspeicherung, die primär eine Klarschrift zum Ziel hatte, die aber auch sicher und einfach maschinell lesbar ist. In diesem Bereich gibt es einige Definitionsschwierigkeiten. Selbstverständlich legen Puristen großen Wert darauf, nur ein normales alphanumerisches Drucksymbol als echte Klarschrift zu bezeichnen, während Kompromißbereite dazu auch Typensätze wie OCR (*Optical Character Recognition* = optische Zeichenerkennung; OCR-Schrift kann von speziellen Geräten direkt gelesen werden) zählen würden, obwohl das Layout der Zeichen so gewählt ist, daß auch mit einigen Abstrichen an die menschliche Lesekultur eine zuverlässige Maschinenlesung gesichert ist. Dabei wurden die OCR-Zeichen so gestaltet, daß Fehlinterpretationen, wie sie durchaus bei verschiedenen Druckbildern möglich sind, vermeidbar werden.
Daß bestimmte Halbleiterspeicher (EPROMs) die erwünschte Eigenschaft haben, unter ultravioletter Bestrahlung ihren Speicherinhalt zu verlieren, ist bekannt. Weniger diskutiert, außer unter den Betroffenen, sind die Risiken für den Datenverlust, abhängig von Logik und Umwelteinflüssen, auf anderen Speichermedien. Unter anderem bestehen hier folgende Möglichkeiten:

- Datenverlust durch Spannungsausfall bei Halbleiterspeichern,
- Kopfdefekt bei Plattenspeichern mit mechanischen und elektrischen Aspekten,
- Bandriß bei Magnetband und Kassetten,
- Datenverlust bei Spool-Routinen (angeblich nicht möglich – passiert trotzdem),
- Schreibfehler auf Band oder Kassette ohne intelligente Hinterbandkontrolle,
- Datenverlust auf Magnetmedien nach oder sogar während der Aufzeichnung durch starke Magnetfelder oder Hf-Einstrahlung.

Diese Auswahl zeigt nur einen kleinen Ausschnitt der tatsächlichen Schwierigkeiten, die beim Erfassen von Daten entstehen können.

Optisch lesbare Daten

Noch vor der Absicht, optisch zu lesen, stand der Wunsch, ein zuverlässiges Speichermedium zu finden. Es sollte nicht flüchtig sein, ausreichende Informationsdichte haben und im weitesten Bereich sicher vor gängigen Umwelteinflüssen sein. Hier sollen unterschiedliche Datenträger keiner Wertung unterzogen werden, aber bei externer Dateneingabe bzw. mobiler Erfassung müssen immer folgende Kriterien berücksichtigt werden:

- codierbare Datenmenge,
- erforderliche Datenmenge,
- Verfahren zur Erstellung des Datenbeleges,
- Eingabeverfahren,
- erforderliche Lesegeschwindigkeit,
- Stabilität des Datenträgers gegenüber Umwelteinflüssen,
- Lesesicherheit einschließlich Prüfmöglichkeit,
- optisch-elektronischer Aufwand zur reinen Signalumsetzung,
- Speicherbedarf für Interpretation und Ablage der Information.

Einen guten Kompromiß stellt in vielen Bereichen der Strichcode dar. Die Mustererkennung (*Pattern Recognition*), also Klarschrift als Code, wäre zwar von der Handhabung her komfortabel, aber auch im Zeitalter des billigen Speicherraums vom Softwareaufwand her immer noch problematisch. Obwohl es mittlerweile Verfahren gibt, alphanumerische Zeichen von verschiedenen Richtungen her zu erkennen und zu interpretieren, liegt die Lesesicherheit nicht im industriell angestrebten Bereich.

Strichcodierungen

Ebenso wie abgespeicherte Daten auf magnetischer Basis beim Überstreichen mit einem Spaltlesekopf ein Bitmuster (also einen logisch erkennbaren

magnetischen Flußwechsel) ergeben, lassen sich gedruckte Hell- und Dunkelzonen optisch lesen und auswerten. Heute werden Strichcodes verwendet, die man grob in folgende Katagorien einteilen kann:

- Acht Strichcodezeilen sind parallel angeordnet und werden gemeinsam gelesen. Der Lesekopf wird von Hand oder automatisch geführt (typische Anwendung: Fotogroßlabors).
- Von einer gesonderten Spur wird ein Taktsignal abgenommen.
- Zur Synchronisation dienen Sonderzeichen, die auch auf einer eigenen Spur untergebracht sein können.
- Sonderzeichen werden farbig dargestellt (typische Anwendung: Pharmaindustrie).

Auffallend sind die Unterschiede der verwendeten Codierungs- und Leseverfahren, die ihre Merkmale durch anwendertypische Anforderungen erhielten. Bei der Lesung von Strichcodierungen in Transportanlagen ist beispielsweise häufig nur eine Leserichtung vorgeschrieben und möglich. Abhängig vom Scanner, der automatischen Abtastvorrichtung, kann auch eine gewisse Winkelabweichung des Symbols bzw. bidirektionales Lesen erforderlich sein oder toleriert werden. Bekannt aus dem Eisenbahnsektor sind auch Ringsymbole, die vom Zentrum oder optischen Nullpunkt aus omnidirektional lesbar sind. Die erreichbare Speicherdichte wird noch an anderer Stelle diskutiert.
Die häufigste und auch wichtigste Methode, Daten durch Strichcodierungen darzustellen, besteht darin, parallele Balken (*Bars*) in solchen Breiten und Abständen anzuordnen, daß bis zu einem Grenzwinkel sogar eine lageunabhängige Lesung möglich ist. Dabei wird eigentlich nur durch eine optoelektronische Vorrichtung eine zeitabhängige Registrierung von hellen und dunklen Zonen durchgeführt. Zu diesem Zeitpunkt hat also noch keine Decodierung im eigentlichen Sinne stattgefunden, aber die Dauer des jeweiligen logischen Zustands für dunkle oder helle Balken kann als logisch 0 oder 1 betrachtet werden (*Bild 1*). Aus Logikzustand und Pulsdauer ergibt sich ein Muster, das ein Rechner interpretieren kann.
Die gängigen Auswerteelemente dafür sind üblicherweise Mikrocomputer, die dezentral eingesetzt werden, da Decodier- und Prüfroutinen rechnerintensiv sind. Als gängigste Variante für die optische Abtastung einschließlich

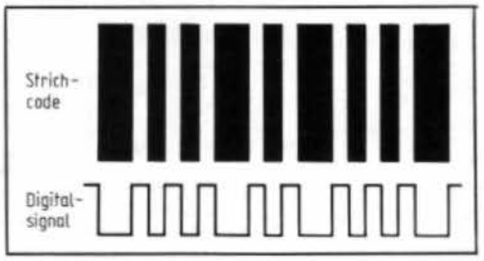


Bild 1. Das Abtastelement liefert ein Digitalsignal, das der Schwärzung des Datenträgers entspricht. Für die Erkennung eines Zeichens ist die Dauer der Pegel maßgebend – im einfachsten Fall stellt ein schmaler Strich das Datenbit 0 dar, während ein breiter Strich 1 repräsentiert. Bei manchen Codes spielt aber auch der Zwischenraum eine Rolle

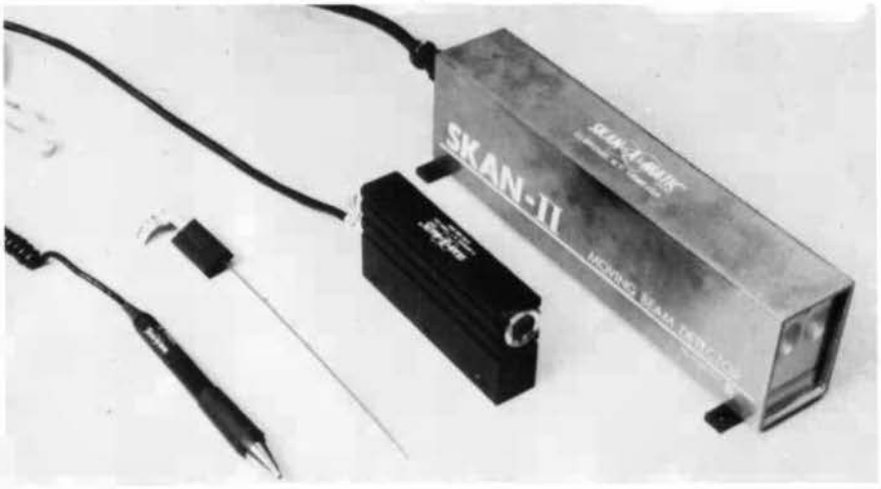


Bild 2. Abtastelemente in verschiedenen Ausführungen. Von links nach rechts: Lesestift für Handeingabe, koaxialer Reflextaster für kritische Umweltbedingungen (z. B. Halbleiterfertigung), Reflextaster für mittlere Entfernungen (bis etwa 120 mm), X/Y-Scanner für Abstände bis etwa 130 mm

der Umsetzung auf elektrische Signale werden folgende Scaneinrichtungen verwendet:

- starre Reflextaster zur Lesung von bewegten Codesymbolen in geringer Entfernung,
- Teleoptiken für mittlere Abstände,
- Laserscanner mit Auslenkung des Abtaststrahls in X- und Y-Richtung durch Drehspiegel bzw. Winkelspiegel,
- Passiver X/Y-Scanner ohne Laser, aber mit dynamischer X/Y-Abtastung,
- Lichtstifte zur Abtastung von Symbolen im unmittelbaren Kontakt oder auch in kleinen Abständen (*Bild 2*).

Aufbau von Strichcodesymbolen

Ein Strichcodesymbol enthält optisch lesbare Daten. Sowohl Inhalt als auch Menge orientieren sich nach dem einzelnen Anwendungsfall. Abhängig davon müssen folgende Punkte berücksichtigt werden:
- Wieviel Bytes soll das Zeichen enthalten?
- Soll es von beiden Seiten lesbar sein?
- Ist eine Prüfziffer erforderlich?
- Welche Start-Stop-Bedingung wird gewählt?
- Sind aus Gründen der Fälschungssicherheit Kombinationen von mehreren Zeichensätzen erforderlich?

Die Anwendungsbereiche der Produktionssteuerung oder Warenflußkontrolle erfordern ebenso wie interne Lagererfassung normalerweise lediglich großen numerischen Inhalt, gekoppelt mit guter Lesesicherheit. Anders sind die Strichcodierungen des Einzelhandels wie EAN oder UPC zu bewerten. Diese Symbole sollen sicher an Datenkassen lesbar sein, außerdem muß jede Manipulation richtiges Lesen verhindern oder beim Lesen die Prüfziffernrechnung beeinflussen.
Strichcodierungen enthalten helle und dunkle Balken, deren Breiten in genau definiertem Verhältnis zueinander stehen, um eine Interpretation zu ermöglichen. Obwohl die meisten Codierungen nur numerische Werte enthalten, gibt es auch alphanumerische Codes. Es ist sogar möglich, den vollständigen ASCII-Zeichensatz inklusive der nicht druckenden Zeichen (Delete, TAB, CR) zu verwenden. Die Logik von Strichcodesymbolen soll so beschaffen sein, daß eine möglichst einfache Zuordnung von Balkenbreite zu BCD-, Hexadezimal- oder ASCII-Zeichen möglich ist. Aus diesem Grunde wird die Dimensionierung der Symbole in definierten Größenverhältnissen zueinander stehen, die direkt aus einer Matrix

hervorgehen. Ein Symbol kann bestimmte Elemente enthalten:

- Randzeichen für Start-Stop und Orientierung,
- Nutzzeichen für den Bytewert,
- Sonderzeichen.

Die Zuordnung zu Matrixwerten wird zum Teil auch als Gewichtung bezeichnet, da jedes einzelne Element einer gewichteten Reihe zugeordnet wird. Zum leichteren Verständnis wird nachfolgend der Aufbau eines einfachnumerischen Codes einschließlich der Spezifikation für die Geometrie vorgestellt.

Beispiel: Skan-A-Matic-2-aus-5-Code

Es handelt sich um einen Code, den die amerikanische Firma Skan-A-Matic eingeführt hat.

Dimensionen
Alle Balkenbreiten stehen im ganzzahligen Teilungsverhältnis zueinander, die kleinste Einheit heißt Modul, der Balken ist schwarz.

1. Schmale Balken sind einen Modul breit (N= Narrow).
2. Breite Balken sind drei Moduln breit (W= Wide).
3. Zwischenräume sind weiß und einen Modul breit.
4. Eine Ziffer wird durch 13 Moduln dargestellt.
5. Die Gesamtbreite eines Randzeichens ist 15 Moduln.
6. Empfohlene Modulbreite minimal 0,254, maximal 0,381 mm.
7. Höhe der Balken 19 mm.
8. Unter jeder codierten Ziffer steht der numerische Wert in Klarschrift.
9. Alle Ziffern im Symbol haben einen Modul Abstand.
10. Das Symbol hat auf allen Seiten eine weiße Schutzzone von 13 mm.

Randzeichen
Es hat die Aufgabe, ein eindeutiges Bitmuster zu erzeugen, das im Prozessor die Startadresse setzt und gleichzeitig eine Seitenerkennung ergibt.
Linkes Randzeichen: WWWNN
Rechtes Randzeichen: WWNNW
Keine der strichcodierten Ziffern enthält drei breite Balken, somit kann ein Randzeichen nie als Ziffer fehlinterpretiert werden. Der Code kann somit von beiden Seiten gelesen werden. *Tabelle 1* zeigt, wie er aufgebaut ist. Jede Ziffer besteht aus einer Kombination von zwei breiten und drei schmalen Balken. Die hellen Zwischenräume enthalten keine Information.

Tabelle 1: Aufbau eines Zeichens beim 2-aus-5-Code (Skan-A-Matic)

Ziffer	Bitwertigkeit				
	1	2	4	7	P*
0	N	N	W	W	N
1	W	N	N	N	W
2	N	W	N	N	W
3	W	W	N	N	N
4	N	N	W	N	W
5	W	N	W	N	N
6	N	W	W	N	N
7	N	N	N	W	W
8	W	N	N	W	N
9	N	W	N	W	N

*Paritätsbit

Prüfziffer

Die Ziffern einer Zahl werden von links nach rechts abwechselnd mit 1 und 3 gewichtet. Als Prüfziffer wird die Zahl angehängt, die nach Addition zur gewichteten Summe ein Vielfaches von 10 ergibt.
Beispiel: Der codierte Inhalt eines Strichcodesymbols ist 10191.
1 x 1 + 0 x 3 + 1 x 1 + 9 x 3
= 29
+ 1 (Prüfziffer)
= 30
Neben der hier vorgestellten Version des 2-aus-5-Codes existieren noch mehrere Varianten. Sie unterscheiden sich vorwiegend in den Randzeichen oder weisen unterschiedliche Prozeduren für die Prüfziffernrechnung auf. Eine Sonderstellung nimmt hier wohl der „Interleaved 2-aus-5-Code" ein. Bei dieser Codierung werden auch variabel breite helle Zonen zwischen den schwarzen Balken als Codierungsträger genutzt. Nach diesem Verfahren lassen sich bei vergleichbaren Balkenbreiten etwa 60% mehr Daten auf der gleichen Druckzone unterbringen. Ähnliche Strukturen weisen die Codes Matrix-2-aus-5 und BCD-2-aus-5 auf.

Strichcodierung im Handel

Im April 1977 wurde in Deutschland beschlossen, im Rahmen der „Europäischen Artikel-Numerierung" (EAN) neue Rationalisierungsmittel in Handel und Warenwirtschaft einzubringen. Dahinter steht einerseits ein Organisationssystem, das Hersteller und Artikel nach einheitlichen Richtlinien mit einer Zahlenkennung belegt. Andererseits wird diese Codierung auch als 8- oder 13stelliges Strichcodesymbol (*Tabelle 2*) verwendet.
Mittlerweile zeigt ein Blick in die Regale eines Supermarktes, wie weit Waren mit der EAN-Strichcodierung versehen sind. Der Handel erwartet von der Codierung mit EAN-Symbolen folgende Verbesserungen:

- schnellere Abwicklung an der Datenkasse, die Artikelnummern decodieren und nach Rückgriff auf die Datenbank als Preis zur Verfügung stellen kann;
- automatische Fortführung des Lagerbestandes über die Rückmeldung des Verkaufs unmittelbar bei Rechnungsstellung;
- selbständige Wiederbestellvorgänge bei Unterschreitung von Mindestbeständen;
- schnelle Inventur mit mobilen Datenerfassungsgeräten, die mit Lesestiften ausgestattet sind;
- Reduzierung von Fehlbedienungen, seien sie beabsichtigt oder unbeabsichtigt.

Die Aufgaben des EAN-Codes liegen in einem völlig anderen Bereich als die der bereits diskutierten 2-aus-5-Codierungen. Da unterschiedliche Druckträger und Druckverfahren zu weiten Abweichungen sowohl bei der Geometrie des Symbols als auch beim Kontrast zwischen hellen und dunklen Balken

Tabelle 2: Logik des 13stelligen EAN-Codes

Zeichen	Zeichensatz A	Zeichensatz B	Zeichensatz C
0	0 0 0 1 1 0 1	0 1 0 0 1 1 1	1 1 1 0 0 1 0
1	0 0 1 1 0 0 1	0 1 1 0 0 1 1	1 1 0 0 1 1 0
2	0 0 1 0 0 1 1	0 0 1 1 0 1 1	1 1 0 1 1 0 0
3	0 1 1 1 1 0 1	0 1 0 0 0 0 1	1 0 0 0 0 1 0
4	0 1 0 0 0 1 1	0 0 1 1 1 0 1	1 0 1 1 1 0 0
5	0 1 1 0 0 0 1	0 1 1 1 0 0 1	1 0 0 1 1 1 0
6	0 1 0 1 1 1 1	0 0 0 0 1 0 1	1 0 1 0 0 0 0
7	0 1 1 1 0 1 1	0 0 1 0 0 0 1	1 0 0 0 1 0 0
8	0 1 1 0 1 1 1	0 0 0 1 0 0 1	1 0 0 1 0 0 0
9	0 0 0 1 0 1 1	0 0 1 0 1 1 1	1 1 1 0 1 0 0

Randzeichen 3 Moduln 101
Trennzeichen 5 Moduln 01010

führen, sind praktisch alle Codierungsparameter vorgegeben (CCG Köln, Der EAN Strichcode, Band2). Nach einer Übergangszeit wird im Handel auch eine rechtliche Verantwortung für die Lesbarkeit des Symbols und der korrekten Codierung wirksam. Die vollständige Beschreibung für den logischen Aufbau einschließlich aller gegebenen Regeln wäre in diesem Rahmen nicht unterzubringen. Die wesentlichen Merkmale sollten doch erwähnt werden.
13stellige EAN-Symbole bestehen aus 12 Nutzzeichen, davon sechs links und sechs rechts, abgesetzt durch ein Trennzeichen. In der linken Seite Zeichensatz A und B mit wechselnder Parität, in der rechten Seite nur Zeichensatz A allein. Das Symbol wird auf beiden Seiten durch ein Randzeichen begrenzt. Die Paritätsfolge in der linken Seite des Symbols ermöglicht die Prüfziffernrechnung. Die Zählung erfolgt von rechts (*Tabelle 3*).

Tabelle 3: Codierung der 13. Stelle mit Vorgabe des zugrunde gelegten Zeichensatzes

Zeichen	Paritätsfolge für linke Seite					
0	A	A	A	A	A	A
1	A	A	B	A	B	B
2	A	A	B	B	A	B
3	A	A	B	B	B	A
4	A	B	A	A	B	B
5	A	B	B	A	A	B
6	A	B	B	B	A	A
7	A	B	A	B	A	B
8	A	B	A	B	B	A
9	A	B	B	A	B	A

A = Zeichensatz A, B = Zeichensatz B

Die Codierung der UPC-Symbole ist dem EAN-Code sehr ähnlich, aber nicht kompatibel. Eines der Probleme bei der Erstellung von EAN- oder UPC-Symbolen ist die Überprüfung von Masterfilmen und gedruckten Symbolen auf logisch richtigen Aufbau, richtig codierten numerischen Inhalt, Prüfziffernrechnung und Einhaltung der zulässigen geometrischen Toleranzen. Das kleinste von elf Codeformaten SC-0 (Scale 0 bis Scale 10) toleriert nur noch eine maximale Abweichung im Druckzuwachs von ± 0,035 mm. Das bedeutet in der Praxis eine definierte Reduzierung des Masterfilms, um abhängig vom Druckverfahren (z. B. Offset, Flexodruck, Buchdruck usw.) mit dem erwarteten Druckzuwachs, also der Balkenverbreiterung trotzdem in die Toleranzzone zu kommen. Für die Messung und Überprüfung all dieser Kennzeichen gibt es rechnergesteuerte

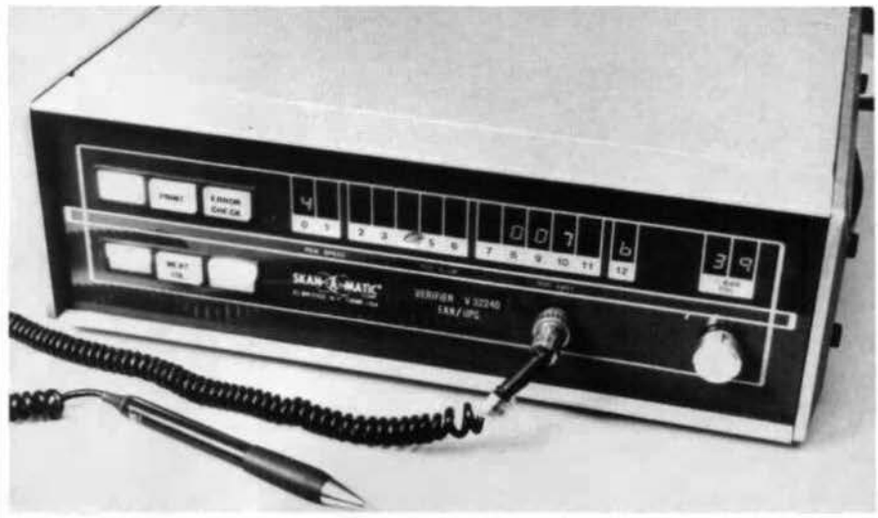


Bild 3. Prüf- und Meßgerät für EAN- und UPC-Code

Systeme, die eben diese Werte verifizieren (*Bild 3*) und ein Meßprotokoll erstellen.
Der Ausdruck des Protokolldruckers zeigt parallel zur Information auf dem Display alle wichtigen Aussagen über das Symbol und dient zum Nachweis der durchgeführten Qualitätskontrolle.

Programmspeicherung über Strichcodes

Papier ist auch in dieser Hinsicht ein geduldiges Medium und durchaus geeignet, Daten als externer Speicher für beliebigen Zugriff verfügbar zu halten. Sofern die Logik des verwendeten Codes es gestatten, die Symbole aneinanderzureihen, ist die Datenspeicherung in Strichcodezeilen ein durchaus gangbarer Weg für schnelle und sichere Dateneingabe. Dies trifft besonders dann zu, wenn Zeichenfolgen in Maschinensprache vorliegen. Bekanntermaßen ist bei Eingabe über die Tastatur ein hoher Konzentrations- und Zeitaufwand erforderlich. Außerdem ist diese Methode fehlerintensiv.

Speicherkapazität von Strichcodesymbolen

Es ist durchaus möglich, mit sehr aufwendigen Optiken und präzise erstellten Symbolen bis zu 10 Zeichen auf 10 mm unterzubringen und auch wieder zu lesen. Allerdings führt eine Änderung von Strichbreiten zum Teil auch durch Verschmutzung des Druckträgers zu lästigen Fehllesungen. Als günstiger Kompromiß erscheinen hier der EAN-Code in nomineller Größe oder der Interleaved 2-aus-5-Code. Beide enthalten abgerundet 4 Zeichen in 10 mm. Eine Zeile kann also etwa 560 Bit enthalten.

Zeitfaktor bei Strichcodierung

Handgeführte Abtastelemente werden üblicherweise mit maximalen Abtastgeschwindigkeiten von 1000 mm/s spezifiziert. Daraus resultiert bei den besprochenen Informationsdichten ein Pegelwechsel von 1600 Hz. Beim EAN-Code oder Interleaved 2-aus 5-Code sind nach etwa 50 Pegelwechseln alle Zeichen inklusive Start und Stop bereits abgefahren. Rund 4 mm nach dem letzten Übergang von einem dunklen Balken auf die helle Schutzzone ist das System bereits voll durch alle Rechneraktivitäten gelaufen, der graphische Wert liegt als decodiertes Zeichen an. Automatische Scannerkassen mit X/Y-Laserscannern liegen in ihren Abtastfrequenzen wesentlich höher, denn der Lesevorgang muß selbst bei schnell bewegten Artikeln sicher sein. Die schnellsten derzeit auf dem deutschen Markt angebotenen Systeme tasten mit ca. 25 m/s ab.

Hardware

Die Hardware zur Lesung von Strichcodierungen hat den Arbeits- und Umweltbedingungen zu entsprechen, um der gestellten Aufgabe mit der erwarteten Leistung gerecht zu werden. Der Bereich beginnt an der untersten Grenze mit kleineren Hobbyrechnern, deren Programmbibliothek sporadisch erneuert wird, jeweils nach Erscheinen einer neuen Fachzeitschrift oder nach Austausch mit anderen Besitzern. Hier kann ein einfach aufgebauter Lesestift aus Plastik mit simpler Optik völlig ausreichend sein. Ein Fahrverkäufer mit mobilem Datenterminal erwartet mehr Strapazierfähigkeit und höhere Verfügbarkeit. Die schwierigsten Bedingungen herrschen im industriellen

Bereich, wo die Funktion 24 Stunden am Tage erwartet wird.

Optische Sensoren

Die Auswahl des geeigneten optischen Sensors muß nach Aufbau des Symbols, Größe, Qualität des Kontrasters und optischer Auflösung getroffen werden.

Lesestift

Für industriellen Einsatz oder Datenkassen werden sogenannte „Code Pens" mit Leichtmetallkörpern (*Bild 2*), Saphiroptiken und Lichtquellen ausgestattet, die entweder im sichtbaren Bereich liegen oder im nahen IR-Bereich bei ca. 950 nm emittieren. Dabei müssen auch die Einsatzorte berücksichtigt werden. Die Standzeiten an automatischen Waagen oder in der Automobilindustrie sind z. B. deutlich kürzer als in anderen Industriezweigen. Das optische Blickfeld liegt zwischen 0,15 und 0,25 mm. Strichcodesymbole, die mit Matrixdruckern gedruckt wurden, können, bedingt durch mechanische Toleranzen, Versatz oder feine helle Lücken im dunklen Umfeld, Probleme aufweisen. Bei zu klein gewähltem Blickfeld können diese Lücken zu unerwünschten Effekten führen.
Lesestifte nehmen im Durchschnitt 120 mA auf, bei 5 V Betriebsspannung. Am Ausgang kann entweder der analoge Fotostrom von etwa 20 (schwarzer Balken) bis 100 µA abgegriffen werden, oder ein interner Vorverstärker stellt ein Spannungssignal zur Verfügung.

Koaxiale Reflextaster

Sie tasten sehr kleine Strichcodesymbole auch in kritischer Umgebung ab (Temperatur, Vibration usw.), obwohl sie nur einen Durchmesser im Millimeterbereich haben. Auf diese Weise kann durch plastische Formung des gefaßten Bündels das Lichtsignal auch um Ecken geführt werden. Die Spitze des Elements verträgt Temperaturen bis zu 125 °C ohne Kühlung, das lichtempfindliche Halbleiterelement sitzt geschützt in meterweitem Abstand (*Bild 2*).

Telescanner

Abstände bis zu 100 mm und Auflösungen von 0,15 mm erlauben zuverlässiges Lesen auch unter schwierigen Bedingungen. Speziallichtquellen (1 A bei 5 V) und Optiken mit geteiltem Strahlengang (Split Beam) erlauben die Abtastung von sichtbaren oder speziellen IR-absorbierenden Codes.

Bild 4. Das sogenannte intelligente Herz eines Strichcode-Auswertesystems – ein Mikrocomputer mit dem Z80 als CPU

X/Y-Scanner

Dynamische Abtastung bis zu 130 mm Abstand mit 100 Durchgängen/s erlauben die „*Moving Beam Scanner*". Die vielfache Lesung erfordert eine spezielle Anpassung in der Decodiersoftware, um innerhalb eines Zeitfensters, das z. B. durch Lichtschranken gesetzt wird, nur *eine* richtige vollständige Lesung an der Schnittstelle zu übergeben. So lassen sich Symbole auf bewegten Behältern oder sogar unmittelbar nach dem Druck auf der laufenden Druckbahn lesen oder prüfen (*Bild 2*).

Funktionseinheiten

Die gelesenen Daten müssen schließlich ausgewertet werden. Sinnvoll ist es, alle Schritte von modularen Funktionseinheiten ausführen zu lassen. Dabei lassen sich Systemänderungen und Erweiterungen einfach und schnell durchführen.
Industrielle Strichcode-Lesesysteme bestehen üblicherweise aus fünf oder sechs Funktionsplatinen, die innerhalb eines Standardgehäuses auf einer Grundplatine gesteckt sind. Ein Beispiel ist das System NEMA 12/14, bei dem ein stabilisiertes Netzteil mit Filterung alle Spannungen für den Analog- und Digitalteil liefert. Das sogenannte intelligente Herz (*Bild 4*) der Anlage ist ein Z80-Prozessorsystem mit residenter Firmware. Es übernimmt die A/D-Umsetzung, die Decodierung und die gesamte Systemsteuerung. An seinem Ausgang liefert es serielle asynchrone ASCII-Zeichen. Strichcodeleser bzw. Mikroprozessorkarte können auf Wunsch auch parallele oder gemultiplexte Daten ausgeben.

Weitere Funktionseinheiten:

- Interfacekarte für Anpassung der Baudrate:
  Gegebenenfalls wird hier auch bei Vielfachleseystemen, die im Verband arbeiten, eine Stationskennung gesetzt, die von 0...99 reicht.
- Displaykarte:
  Ansteuerteil für 7-Segment-LED-Anzeige.
- „*Party-Line*"-Computer-Interface:
  Decodierte Daten mit Stationskennung und definierter Annahmepriorität werden von dieser speziellen Funktionskarte verwaltet, überprüft und durch Optokoppler getrennt auf TTY (20-mA-Stromschleife) umgesetzt.
- Tastenfelder:
  Zusatzinformationen, die nicht in codierter Form vorhanden sind, können in variabler Länge gemeinsam mit dem nächsten codierten Wert und der Stationskennung als gemeinsamer Datensatz übergeben werden.
- Drucker für Strichcodes:
  Neben den Produktcodes auf Verpakkung, deren Masterfilm fotografisch hergestellt wird, ist in vielen Fällen die Einzelerstellung bei fortlaufender Ziffernfolge sinnvoll. Dabei verwendet man entweder Typenraddrucker mit Strichcodesymbolen oder Matrixdrucker. Nach jüngsten Erfahrungen hat auch ein deutscher Hersteller, die Mannesmann-Tally GmbH ein recht geeignetes Gerät auf den Markt gebracht (*Bild 5*). Der M132/99C beherrscht mittlerweile an die zehn verschiedene Codelogiken und zeigt eine erfreuliche Druckqualität.

  Die wichtigsten Daten sind:
  - 9 x 9-Matrix,
  - Matrixkopf für OCR-A/B-Barcodes,
  - Interne Mikroprozessorsteuerung,
  - V-24-Schnittstelle (asynchron).

Software

Die Fähigkeiten der Mikroprozessoren Z80 und 8086 boten sich geradezu für diese Anwendung an. Deshalb wurden beide Typen für die geschilderten Applikationen vielfach ausgewählt.
Der Programmaufwand, zeitstabile TTL-Signale mit einer Ereignismatrix zu vergleichen, wäre relativ gering. Die praktische Seite zeigt allerdings etwas unterschiedliche Bedingungen. Da eine gleichmäßige Abtastung kaum realisierbar ist, gilt es mit einigen Schwierigkeiten fertig zu werden.
Beim manuellen Abtasten sollten lineare Lesegeschwindigkeiten zwischen 8 und 100 cm/s zulässig sein. Außerdem darf ein Beschleunigen oder Verzögern in der Größenordnung von 30% innerhalb des gesamten Lesevorganges das Decodieren nicht erschweren. Typisch sind folgende Werte:

30 cm/s ± 30%,
60 cm/s ± 30%,
13 cm/s ± 30%.

Diese Werte können noch einmal Einflüsse durch den Abfahrweg im Symbol erfahren, denn es ist nicht ausgeschlossen, daß beim Abtasten mit dem Lichtstift ein leicht bogenförmiger oder diagonaler Weg beschrieben wird.
Je nach Effektivität der Software muß bei Decodierung eines einfachen numerischen Codes mit einem Speicherbedarf von mindestens 2 KByte gerechnet werden. Abhängig von den zulässigen Zeitvariablen kann ein Zuschlag erforderlich sein. Folgende Überlegungen und Aktionen sind in jedem Fall erforderlich:

- Erkennung des Startsignals als Symbolbegrenzung links oder rechts,
- Erkennung des „Stop"-Signals,
- Ablage auf Register,
- Erkennung von Trennzeichen, Teilung der Summe aller Nutzzeichen,
- Ablage auf Register,
- Vergleich mit Logikmatrix nach Ereignisfällen,
- Zuordnung zu ASCII-Zeichen,
- Erkennung von Zeichensatz mit Parität gerade/ungerade,
- Zuordnung der Paritätsinformation nach verwendetem Zeichensatz,

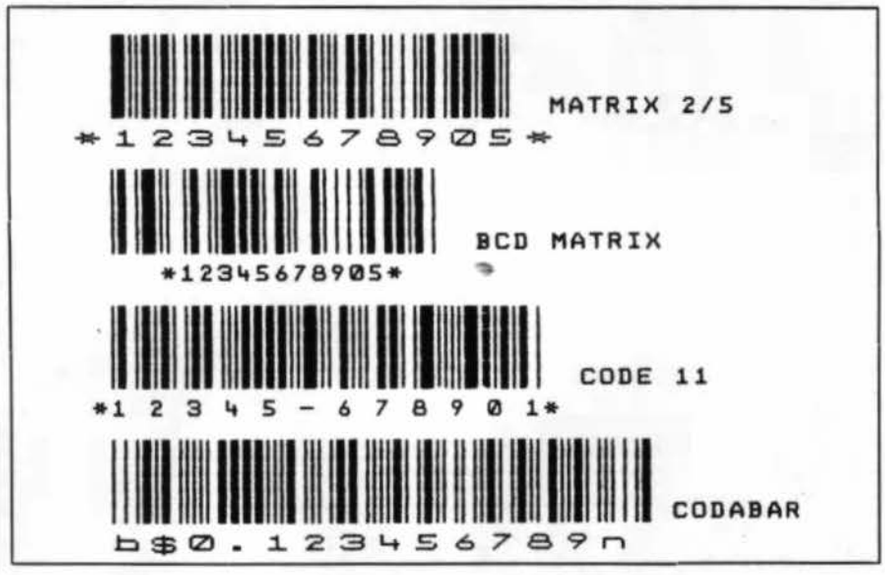


Bild 5. Druckbeispiele, erstellt mit dem Matrixdrucker M132/99C (Mannesmann-Tally)

- Prüfung auf Querparität für Prüfzifferrechnung,
- Rückgriff auf Start-Stop und Einordnung auf richtige Dezimalfolge, Zuordnung des Trennzeichens.

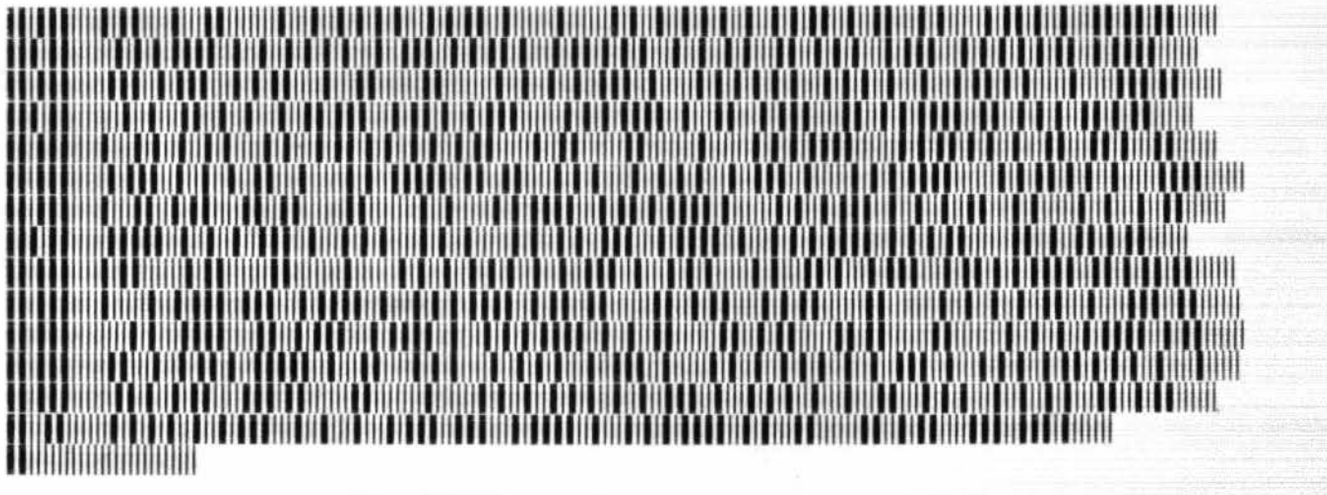
Außer an diesen groben Aktionskriterien soll noch an bestimmte Optimierungsroutinen erinnert werden, die es erlauben, gestaffelte Zeitauflösungen einzuführen und somit die Interpretationssicherheit beeinflussen.
Genauso gehen auch abhängig von der verwendeten Codelogik die Teilerverhältnisse in die Softwareüberlegung mit ein.

Literatur

[1] Choosing Optical Bar Code Scanners. Druckschrift der Fa. Skan-A-Matic.

Strichcode-Programm: Primfaktor-Zerlegung in Basic

Das hier abgedruckte Strichcode-Programm ist in der üblichen Basic-Syntax heutiger Tischcomputer (PET, AIM, TRS-80 usw.) geschrieben und zerlegt eine eingegebene Zahl in ihre sämtlichen Primfaktoren. Das verwendete Datenformat entspricht der in diesem Heft beschriebenen Norm, d. h. die hier acht Basic-Programmzeilen mit je etwa 30 Zeichen werden als Folge von ASCII-Zeichen in den Strichcode umgewandelt. Übrigens: Wer keinen Strichcode-Leser besitzt, findet das Programm auch in dem im Franzis-Verlag erschienenen Buch „Basic für Mikrocomputer".



Dr. Harald Lindner

Basic mit Labels

In Heft 22/1980 der FUNKSCHAU stellte Hans-Georg Joepgen die Frage, ob es denn nicht auch in Basic möglich sei, Programmsprünge nicht mit absoluten Zeilennummern, sondern mit symbolischen Labels auszuführen, etwa wie GOTO "DRUCK". Obwohl nur wenige Basic-Interpreter über solchen Komfort verfügen, läßt sich das Problem leicht mit einem kleinen Maschinenprogramm lösen.

```
7ECF A9 4C     LDA #4C        Veraenderung der Leseroutine:
7ED1 85 79     STA 79         Der Sprungbefehl JMP 7EDC
7ED3 A9 DC     LDA #DC        wird eingefuegt
7ED5 85 7A     STA 7A
7ED7 A9 7E     LDA #7E
7ED9 85 7B     STA 7B
7EDB 60        RTS
7EDC C9 8B     CMP #8B        'IF'-Code?
7EDE F0 0C     BEQ 7EEC       Ja: ->
7EE0 C9 89     CMP #89        'GOTO'-Code?
7EE2 F0 08     BEQ 7EEC       Ja: ->
7EE4 C9 8D     CMP #8D        'GOSUB'-Code?
7EE6 F0 04     BEQ 7EEC       Ja: ->
7EE8 C9 91     CMP #91        'ON'-Code?
7EEA D0 15     BNE 7F01       Nein: ->
7EEC 48        PHA            Code merken
7EED 8A        TXA            X-Register
7EEE 48        PHA            ebenso merken
7EEF BA        TSX            Zur Pruefung der Ruecksprungadresse
7EF0 BD 04 01  LDA 0104,X     Ist es C6F9 (Interpreter-Routine)?
7EF3 C9 C6     CMP #C6
7EF5 D0 07     BNE 7EFE       Nein: ->
7EF7 BD 03 01  LDA 0103,X
7EFA C9 F9     CMP #F9
                                                          ▶
```

Bild 1. Das ist das Maschinenprogramm im Disassembler-Format, welches Computern des Typs CBM 3032 (32 KByte) gestattet, nach den Sprungbefehlen GOTO und GOSUB nicht nur absolute Zeilennummern, sondern auch symbolische Namen zu verwenden. Basic-Programme werden dann wesentlich übersichtlicher. Das Maschinenprogramm liegt am oberen Ende des 32-KByte-RAM; bei weniger Speicherplatz ist eine Adressenverschiebung erforderlich

In der Programmiersprache Basic sind für Sprünge (GOTO) und Unterprogrammaufrufe (GOSUB) Zeilennummern erforderlich. Viel schöner sind die Möglichkeiten höherer Sprachen wie Fortran, Pascal usw., Programmzeilen mit Labels, d. h. Namen zu versehen. Beginnt ein Unterprogramm z. B. in der Programmzeile 1234, so kann man es mit einem Namen versehen, etwa FEHLERCHECK, und kann es mit GOSUB"FEHLERCHECK" statt mit GOSUB1234 aufrufen. Genau diese Möglichkeit bietet das Maschinenprogramm in *Bild 1* für die CBM-Computer der Serie 3001. Durch eine Änderung der Leseroutine sind nun nach GOTO und GOSUB auch Labels erlaubt (nicht aber nach RUN, GO TO, THEN und LIST). Labels werden an den Anfang einer Programmzeile nach REM eingetragen. Durch das 'REM' werden sie beim normalen Ablauf des Programms überlesen. Ein Beispiel:

```
200GOSUB"DRUCKE"
201END
300REMDRUCK
301END
400REMDRUCKEN
401PRINT"ABC"
402RETURN
500REMDRUCKE
501PRINT"ENDE"
502RETURN
```

ergibt nach RUN den Ausdruck ABC. Durch GOSUB"DRUCKE" wird das Programm von Anfang an nach dem Label DRUCKE durchsucht. Dabei wird in der Zeile 400 das Label DRUKKEN gefunden, das in den ersten Buchstaben mit dem gesuchten Label über-

```
7EFC F0 0A     BEQ 7F08        Ja: ->
7EFE 68        PLA
7EFF AA        TAX             X-Register wiederherstellen
7F00 68        PLA             Akkumulator ebenso
7F01 C9 3A     CMP #3A         Zur Pruefung ob Ziffer
7F03 B0 D6     BCS 7EDB        >? Ja: -> (RTS)
7F05 4C 7D 00  JMP 007D        Leseroutine normal fortsetzen
7F08 A9 C3     LDA #C3         Ruecksprungadresse L
7F0A 9D 03 01  STA 0103,X      korrigieren
7F0D 68        PLA             X-Register
7F0E AA        TAX             wiederherstellen
7F0F 68        PLA             gemerkten Code ebenso
7F10 C9 8B     CMP #8B         'IF'-Code?
7F12 D0 30     BNE 7F44        Nein: ->
7F14 20 A7 CC  JSR CCA7        'IF'-Bedingung auswerten
7F17 20 76 00  JSR 0076        Letztes Zeichen holen
7F1A C9 89     CMP #89         'GOTO'-Code?
7F1C F0 1F     BEQ 7F3D        Ja: ->
7F1E A9 A7     LDA #A7         'THEN'-Code
7F20 20 FA CD  JSR CDFA        Wenn nicht THEN-Code: SYNTAX-ERROR
7F23 A5 5E     LDA 5E          Ist Null, wenn IF-Bedingung falsch
7F25 F0 1A     BEQ 7F41        Bedingung falsch? Ja: ->
7F27 20 76 00  JSR 0076        Letztes Zeichen holen
7F2A C9 8B     CMP #8B         'IF'-Code?
7F2C F0 E6     BEQ 7F14        Ja: ->
7F2E C9 91     CMP #91         'ON'-Code?
7F30 F0 16     BEQ 7F48        Ja: ->
7F32 C9 8D     CMP #8D         'GOSUB'-Code?
7F34 F0 3A     BEQ 7F70        Ja: ->
7F36 C9 89     CMP #89         'GOTO'-Code?
7F38 F0 50     BEQ 7F8A        Ja: ->
7F3A 4C 48 C8  JMP C848        IF-Routine normal fortsetzen
7F3D A5 5E     LDA 5E          Ist die IF-Bedingung erfuellt?
7F3F D0 49     BNE 7F8A        Ja: ->
7F41 4C 43 C8  JMP C843        zur REM-Routine
7F44 C9 91     CMP #91         'ON'-Code?
7F46 D0 24     BNE 7F6C        Nein: ->
7F48 20 75 D6  JSR D675        Ausdruck auswerten
7F4B 48        PHA             folgendes Zeichen merken
7F4C C9 8D     CMP #8D         'GOSUB'-Code?
7F4E F0 04     BEQ 7F54        Ja: ->
7F50 C9 89     CMP #89         'GOTO'-Code?
7F52 D0 14     BNE 7F68        Nein: -> SYNTAX ERROR
7F54 A5 62     LDA 62          Wert des Sprungverteilers
7F56 85 7C     STA 7C          merken
7F58 C6 7C     DEC 7C          und vermindern
7F5A F0 0F     BEQ 7F6B        =0? Ja: ->
7F5C 20 A7 CC  JSR CCA7        Ausdruck auswerten
7F5F 20 76 00  JSR 0076        letztes Zeichen holen
7F62 C9 2C     CMP #2C         Komma-Code?
7F64 F0 F2     BEQ 7F58        Ja: ->
7F66 68        PLA             Code wiederherstellen
7F67 60        RTS             zur Interpreter-Routine
7F68 4C 03 CE  JMP CE03        SYNTAX ERROR melden
7F6B 68        PLA             GOTO/GOSUB-Code wiederherstellen
7F6C C9 89     CMP #89         'GOTO'-Code?
7F6E F0 1A     BEQ 7F8A        Ja: ->
7F70 A9 03     LDA #03         Zur Pruefung ob genuegend
7F72 20 1B C3  JSR C31B        Platz im Stapel vorhanden ist
7F75 A5 78     LDA 78          Programmzeiger H
7F77 48        PHA             auf den Stapel legen
7F78 A5 77     LDA 77          Programmzeiger L
7F7A 48        PHA             und
7F7B A5 37     LDA 37          Zeilennummer H
7F7D 48        PHA             und
7F7E A5 36     LDA 36          Zeilennummer L
7F80 48        PHA             und
7F81 A9 8D     LDA #8D         'GOSUB'-Code
7F83 48        PHA             und
7F84 A9 C6     LDA #C6         Ruecksprungadresse H
7F86 48        PHA             und
7F87 A9 C3     LDA #C3         Ruecksprungadresse L
7F89 48        PHA             ebenso auf den Stapel legen
7F8A 20 A7 CC  JSR CCA7        Ausdruck auswerten
7F8D A5 07     LDA 07          Flag fuer Variablentyp
7F8F 30 0C     BMI 7F9D        String? Ja: ->
7F91 20 D2 D6  JSR D6D2        Zeilenadresse holen
7F94 A5 12     LDA 12          Zeilenadresse H
7F96 C9 FA     CMP #FA         mit Hoechstwert vergleichen
7F98 B0 CE     BCS 7F68        zu gross: ->
7F9A 4C B0 C7  JMP C7B0        GOTO-Routine mit Zeilennummer
7F9D 20 7D D5  JSR D57D        Stringadresse und -laenge holen
```

▶

einstimmt. Das Label DRUCKE in der Zeile 500 wird deshalb nicht gefunden.

Neben diesen neuen Möglichkeiten sind natürlich die alten Befehle wie GOTO 500 usw. weiter zu verwenden. Alte Basic-Programme laufen daher im wesentlichen wie gewohnt. Geringe Unterschiede sind zwingend vorhanden: normalerweise ergäbe
ON 3 GOTO 9, "A", 5
einen Syntax-Error, ist aber jetzt zulässig.

Den Befehl ON I GOTO... kann man nun auch ersetzen durch GOTO A$(I), wenn man die Sprungziele in dem Stringfeld A$(I) abgespeichert hat.

Auch im Dialog mit dem CBM ergeben sich neue Möglichkeiten: durch INPUT A$: GOTO A$ kann man Teile im Programm mit ihrem Namen nennen und dort fortfahren.

Beim Austesten des Programms ergab sich noch eine Macke des CBM Basic-Interpreters: ist die Zeilennummer 0 im Basic-Programm vorhanden, so bewirkt eine RUN/GOTO/GOSUB-Anweisung, auf die keine Ziffer folgt, einen Sprung zur Zeile 0. Richtig wäre dagegen die Meldung „Syntax error". Dieses Fehlverhalten wird für GOTO und GOSUB durch das Maschinenprogramm in Bild 1 korrigiert. Übrigens listet LIST-0 auch nicht bis 0, sondern bis zum Ende des Programms.

Es gibt zwei neue Fehlermeldungen: GOTO Z$ ergibt einen STRING TOO LONG ERROR, wenn Z$ mehr als 250 Zeichen (dezimal) enthält. Wird ein Label nicht gefunden, so meldet der CBM einen LABEL NOT FOUND ERROR. Für Labels sind alle Zeichen erlaubt. Man sollte dabei besonders auf Leerstellen achten: GOTO"A" findet z. B. 80REMA, nicht aber 80REM A! GOTO" " ist auch zulässig und findet die erste mit REM beginnende Zeile im Programm.

Wie dieses Maschinenprogramm für den CBM zeigt, ist die Programmiersprache Basic durchaus zur Verarbeitung von Labels geeignet. Eine Erweiterung von Basic durch neue Befehle, wie H.-G. Joepgen vorschlug, ist unnötig. Stattdessen sind lediglich die Routinen für GOTO usw. im Basic-Interpreter des jeweiligen Computers geeignet zu wählen. In neueren Computern

wie dem Basic-Taschenrechner von Sharp ist dies schon implementiert.

Wir haben bisher nach GOTO und GOSUB Ausdrücke vom Typ „String" betrachtet. Zusätzlich sind nach diesen beiden Befehlen auch numerische Ausdrücke erlaubt; dadurch können berechnete Sprünge durchgeführt werden. Ein Beispiel: für I=10 wirkt GOTO7+I×I wie GOTO107, springt also nach der Zeile mit der Zeilennummer 107.

Durch die Änderung der Leseroutine mit dem Maschinenprogramm in Bild 1 wird der Basic-Interpreter etwas langsamer. Außerdem kann dieses Maschinenprogramm auch nicht zusammen mit anderen Programmen eingesetzt werden, die die Leseroutine ebenfalls verändern, z. B. Toolkit, DOS usw. Auf Labels und berechnete Sprünge braucht man aber auch in diesem Fall nicht zu verzichten, da die Routinen für GOTO (7F8A hex, 32650 dez) und GOSUB (7F70 hex, 32624 dez) direkt aufgerufen werden können. Dafür ist nur der Teil 7F70 bis 7FFF des Programms in Bild 1 erforderlich. In 7FA9 und 7FDA muß dann hex 62 stehen. Die Aufrufe erfolgen dann z. B. durch SYS32650,I bzw. SYS32624,Z$ anstelle von GOTO I bzw. GOSUB Z$. Bild 2 gibt das Maschinenprogramm als Basic-Poke-Routine wieder.

```
7FA0 18        CLC
7FA1 69 05     ADC #05        zur Stringlaenge 5 addieren
7FA3 90 03     BCC 7FA8       Summe<100 (hex)? Ja: ->
7FA5 4C 32 D5  JMP D532       Nein: STRING TOO LONG ERROR
7FA8 85 7C     STA 7C         Summe merken
7FAA 8A        TXA            Stringadresse L
7FAB E9 04     SBC #04        Fuer Labelsuche 5 abziehen (Carry=0)
7FAD 85 11     STA 11         und merken
7FAF 98        TYA            Stringadresse H
7FB0 E9 00     SBC #00        Carry beruecksichtigen
7FB2 85 12     STA 12         und Ergebnis merken
7FB4 A5 28     LDA 28         Anfangszeiger L Basic-Text
7FB6 A6 29     LDX 29         Anfangszeiger H Basic-Text
7FB8 85 5C     STA 5C         als Suchzeiger L merken
7FBA 86 5D     STX 5D         Suchzeiger H ebenso
7FBC A0 01     LDY #01        Offset fuer
7FBE B1 5C     LDA (5C),Y     Adresse H naechste Basic-Zeile
7FC0 F0 27     BEQ 7FE9       Ende Basic-Text erreicht? Ja: ->
7FC2 AA        TAX            neue Zeilenadresse H merken
7FC3 A0 04     LDY #04        Offset fuer 1. Zeichen der Basic-Zeile
7FC5 B1 5C     LDA (5C),Y     Zeichen in den Akkumulator laden
7FC7 C9 8F     CMP #8F        'REM'-Code?
7FC9 F0 0D     BEQ 7FD8       Ja: -> zum Labelvergleich
7FCB A0 00     LDY #00        Offset fuer
7FCD B1 5C     LDA (5C),Y     neue Zeilenadresse L
7FCF 4C B8 7F  JMP 7FB8       zur Pruefung der naechsten Zeile
7FD2 B1 11     LDA (11),Y     Zeichen aus Sprungziel
7FD4 D1 5C     CMP (5C),Y     mit Zeichen aus Label vergleichen
7FD6 D0 F3     BNE 7FCB       verschieden? Ja: weitersuchen!
7FD8 C8        INY            Offset erhoehen fuer naechstes Zeichen
7FD9 C4 7C     CPY 7C         Sind alle Zeichen geprueft?
7FDB D0 F5     BNE 7FD2       Nein: weiter vergleichen!
7FDD 18        CLC            Ja: Label gefunden
7FDE A0 01     LDY #01        Startadresse der folgenden Zeile
7FE0 20 51 C5  JSR C551       nach (A/X) bringen
7FE3 86 5D     STX 5D         Adresse H der Folgezeile
7FE5 38        SEC            Programmzeiger auf Zeichen vor
7FE6 4C CF C7  JMP C7CF       der naechsten Zeile einstellen
7FE9 A2 06     LDX #06        Zur Meldung von "LABEL
7FEB BD F9 7F  LDA 7FF9,X     NOT FOUND ERROR"
7FEE 20 D2 FF  JSR FFD2       Zeichen drucken
7FF1 CA        DEX            Zeiger vermindern
7FF2 10 F7     BPL 7FEB       >=0? Ja: weiterdrucken!
7FF4 A0 28     LDY #28        Offset fuer "NOT FOUND ERROR"
7FF6 4C 7D F5  JMP F57D       Fehlermeldung drucken
7FF9 4C 45 42 41 4C 3F 0D     'L E B A L ? CR'
```

```
100 REM ### SPRUENGE MIT DEM CBM ###
110 E=256*PEEK(53)+PEEK(52)-1:A=E-304
120 FORI=ATOE:READX:POKEI,X:NEXT
130 FORI=1TO3:GOSUB160:NEXT
140 H=INT(A/256):POKE53,H:POKE52,A-256*H
150 SYSA:NEW
160 READX,Y,Z:X=A+X:Y=A+Y:Z=A+Z:H=INT(Z/256)
170 POKEX,Z-256*H:POKEY,H:RETURN
200 DATA169,76,133,121,169,220,133,122,169,126,133,123,96,201,139,240
210 DATA12,201,137,240,8,201,141,240,4,201,145,208,21,72,138,72
220 DATA186,189,4,1,201,198,208,7,189,3,1,201,249,240,10,104
230 DATA170,104,201,58,176,214,76,125,0,169,195,157,3,1,104,170
240 DATA104,201,139,208,48,32,167,204,32,118,0,201,137,240,31,169
250 DATA167,32,250,205,165,94,240,26,32,118,0,201,139,240,230,201
260 DATA145,240,22,201,141,240,58,201,137,240,80,76,72,200,165,94
270 DATA208,73,76,67,200,201,145,208,36,32,117,214,72,201,141,240
280 DATA4,201,137,208,20,165,98,133,124,198,124,240,15,32,167,204
290 DATA32,118,0,201,44,240,242,104,96,76,3,206,104,201,137,240
300 DATA26,169,3,32,27,195,165,120,72,165,119,72,165,55,72,165
310 DATA54,72,169,141,72,169,198,72,169,195,72,32,167,204,165,7
320 DATA48,12,32,210,214,165,18,201,250,176,206,76,176,199,32,125
330 DATA213,24,105,5,144,3,76,50,213,133,124,138,233,4,133,17
340 DATA152,233,0,133,18,165,40,166,41,133,92,134,93,160,1,177
350 DATA92,240,39,170,160,4,177,92,201,143,240,13,160,0,177,92
360 DATA76,184,127,177,17,209,92,208,243,200,196,124,208,245,24,160
370 DATA1,32,81,197,134,93,56,76,207,199,162,6,189,249,127,32
380 DATA210,255,202,16,247,160,40,76,125,245,76,69,66,65,76,63
390 DATA13,5,9,13,257,258,233,285,286,298
READY.
```

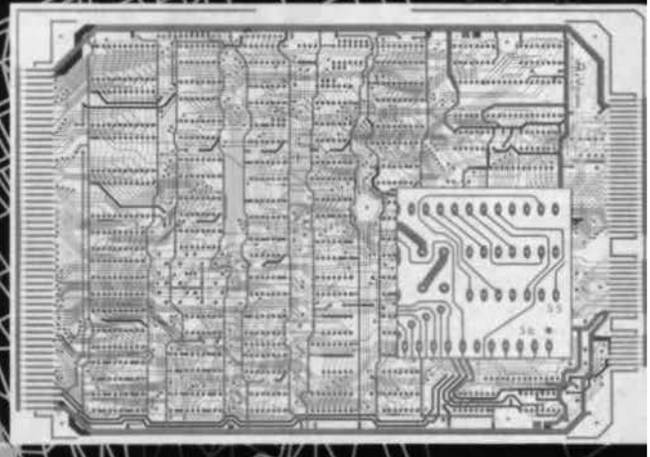
Bild 2. Das Maschinenprogramm aus Bild 1 läßt sich auch als Basic-Programm mit DATA-Statements schreiben, so daß es nach RUN automatisch initialisiert wird. Aber, Vorsicht: Drücken Sie nicht RUN, bevor dieses Programm auf Kassette oder Floppy ist – es löscht sich nämlich mit dem NEW-Befehl in Zeile 150 selbst!

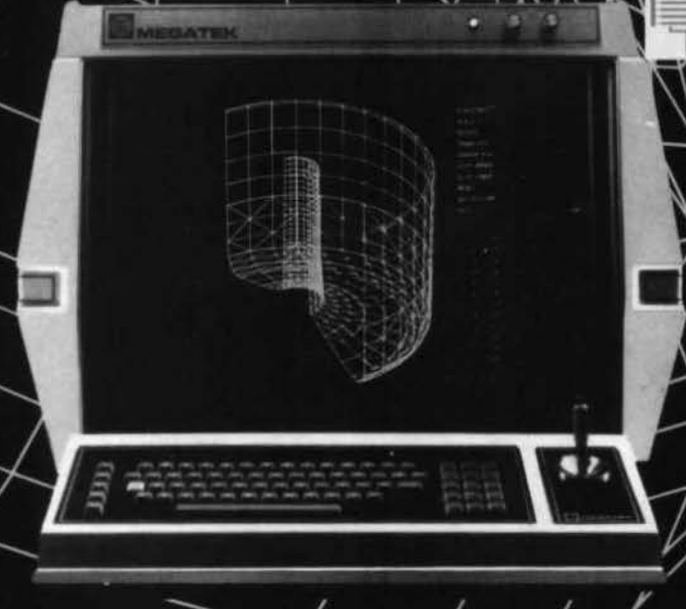
Ideen werden Wirklichkeit.

Zwischen Idee und Wirklichkeit liegen oft Welten.
Ganz gleich wo Sie arbeiten, ob in der Elektronik, im Maschinen- oder Kraftfahrzeugbau, ob Sie entwickeln, planen, konstruieren oder managen, mit der Modellbildung schaffen Sie sich Wettbewerbsvorteile.
Bevor Sie etwas realisieren zeigt Ihnen Ihr Computermodell alle Grenzen, Möglichkeiten und Vorteile. Sie können Ihr Modell interaktiv verändern, drehen, vergrößern oder belasten. Die hohe Auflösung unserer Systeme, die 4 k × 4 k beträgt, bekommt durch Farbe eine zusätzliche Dimension. 16 gleichzeitig darstellbare Farben aus einer Palette von über 4000 geben allen zwei- und dreidimensionalen Modellen ein wesentliches Mehr an Information.
Der einfache Modellbau durch die „Modulare-Software", mit über 20 abrufbaren Charaktersätzen, mit Balken-, Kreis- und Liniendiagrammen, mit einer Vielzahl wählbarer Koordinatensysteme, mit unterschiedlichen Vektortechniken und abrufbaren Symbolen, bildet die idealen Voraussetzungen für schnelle Modellentwicklungen und wirtschaftliche Managemententscheidungen.
Informieren Sie sich über die vielen Einsatzgebiete unserer Megatek-Systeme.

ROHDE & SCHWARZ
RSE
ENGINEERING AND SALES GMBH

Rohde & Schwarz
Engineering and Sales GmbH
Graf-Zeppelin-Straße 18
5000 Köln 90
Telefon (0 22 03) *49-1
Telex 8 873 288 (rse d)





Rudolf Hofer

Apple II liest Strichcode

Um Programme (z. B. in Basic), die im Strichcode abgedruckt sind, in den Computer zu bringen, sind zwei Dinge nötig: der auf Seite 45 vorgestellte Leser und ein kleines Hilfsprogramm.

Wer sich nicht dafür interessiert, wie das Programm in *Bild 1* im einzelnen funktioniert, kann es einfach anhand folgender Anleitung benutzen:

- ☐ Der Leser wird nach *Bild 2* an den sogenannten „Game I/0 Connector" des Apple angeschlossen.
- ☐ Die Daten aus Bild 1 (A9 FF 85...) werden zeilenweise mit Hilfe des Apple-Monitors ab Adresse 3E00 eingetippt und auf Band bzw. Diskette gespeichert. Falls ohne Diskettenbetriebssystem gearbeitet wird, ist das Byte an der Stelle 3E0C (20) durch 60 zu ersetzen.
- ☐ Sollen Daten eingelesen werden, wird folgendes Kurzprogramm gestartet (nicht im Direktmodus eingeben):
  10000 HOME
  10001 HIMEM: 15871
  10002 CALL 15872
  Ab jetzt ist keine Eingabe von der Tastatur aus mehr möglich. An ihre Stelle tritt der Strichcodeleser.
- ☐ Die Strichcodezeilen werden der Reihe nach mit dem Leser abgefahren. Achtung: Etwas vor dem ersten Strich beginnen und nicht vor dem letzten Strich abstoppen! Leser u. U. an Lineal anlegen! Mit etwas Übung geht es aber frei Hand wesentlich flotter. Damit man jedoch mit geringer Geschwindigkeit noch lesen kann, muß die Zeitkonstante in Speicherzelle 3EA7 etwa auf den Wert 19 (hex) geändert werden. Ist eine Zeile richtig gelesen, erscheint der entsprechende Text auf dem Bildschirm, und man kann die nächste Zeile einlesen. Liegt ein Lesefehler vor, dann gibt der Lautsprecher des Apple einen kurzen Piepton ab. In diesem Fall muß der Lesevorgang wiederholt werden. Sind alle Zeichen gelesen, drückt man die Reset-Taste, und die Tastatur ist wieder in Betrieb. Jetzt kann das Programm auf Kassette oder Diskette abgespeichert werden.

Funktion des Einleseprogramms

Das Programm beginnt bei Adresse 3E00 (hex), also am Ende des 16-K-Bereichs. Der vorher erwähnte Befehl CALL 15872 entspricht dem Start an der Stelle INIT. Dieser Programmteil ändert den Inhalt der Speicherzellen 38/39 (hex.), in denen normalerweise die Anfangsadresse der Tastaturabfrageroutine steht. Der neue Wert ist die Anfangsadresse des eigentlichen Einleseprogramms READER, dessen Ablauf in *Bild 3* dargestellt ist. Der Befehl JSR $3EA ist nur mit Diskettenbetriebssystem erforderlich und muß ansonsten entfallen.

```
0800             1  ;CODEREADER
0800             2  ;*********************************
0800             3  XTEMP1 EPZ $F6
0800             4  REF0   EPZ $F7
0800             5  REF1   EPZ $F8
0800             6  REF    EPZ $F9
0800             7  BUF    EPZ $FA
0800             8  CHCKL  EPZ $FB
0800             9  CHCKH  EPZ $FC
0800            10  XTEMP  EPZ $FD
0800            11  PFFLAG EPZ $FE
0800            12  PNT    EPZ $FF
0800            13  KSWL   EPZ $38
0800            14  KSWH   EPZ $39
0800            15  PA     EQU $C061          ;PORT
0800            16  ZEIT   EQU 8
0800            17  BELL1  EQU $FBDD
0800            18  IOSAVE EQU $FF4A
0800            19  IOREST EQU $FF3F
0800            20  ;*********************************
3E00            21         ORG $3E00
3E00 A9FF       22  INIT   LDA #$FF           ;FLAG PUFFER LEER
3E02 85FE       23         STA PFFLAG
3E04 A910       24         LDA #READER
3E06 8538       25         STA KSWL
3E08 A93E       26         LDA /READER
3E0A 8539       27         STA KSWH
3E0C 20EA03     28         JSR $3EA           ;NUR MIT DOS
3E0F 60         29         RTS
```

Bild 1. Listing des Strichcode-Einleseprogrammes für den Apple II. Es ist weitgehend modular aufgebaut. Das Programm nutzt die Tastaturabfrage-Routine des Apple II aus

Anstatt von der Tastatur bekommt der Computer also jetzt von diesem Programm Zeichen für Zeichen geliefert. Es muß lediglich dafür gesorgt werden, daß der Akku das gewünschte Zeichen enthält, daß am Ende der Befehl RTS (*Return from Subroutine*) steht und daß keine anderen Registerinhalte zerstört werden.

Da man die Daten vom Strichcodeleser erst auf ihre Richtigkeit prüfen muß, bevor man sie auf die beschriebene Weise an die „Zeichenschnittstelle" des Computers übergibt, muß man sie erst in einem Puffer ablegen. Dieser Puffer wird so lange neu gefüllt, bis die Prüfsumme stimmt. Erst dann liegen lauter richtige Daten vor. Jetzt kann Zeichen für Zeichen übergeben werden.

Leider ist nicht bei jedem Computer eine solche Zeichenschnittstelle so problemlos zugänglich. Deshalb ist das Rahmenprogramm READER nicht universell verwendbar. Prinzipiell für jede 6502-Maschine ist aber der Teil geeignet, der im Flußdiagramm mit „Zeichen in Puffer einlesen bis o. k." bezeichnet ist. Das Programmsegment ab START liest eine Strichcodezeile und legt die Daten in einem Puffer ab. Es ruft verschiedene Unterprogramme auf und berechnet die Prüfsumme.

Aus dem Apple-Monitor werden BELL1 (Piepton erzeugen) IOSAVE (CPU-Register retten) und IOREST (CPU-Register wieder laden) verwendet. Das Unterprogramm BYTE liest ein Byte ein. Es ruft seinerseits BITTST und AUSW auf. Die letzten beiden Routinen sind der Kern des gesamten Einleseprogramms.

Die wichtigsten Unterprogramme

Die Routine BITTST ist leicht erklärt. Sie zählt im Indexregister X die Dauer eines Schwarzpegels – im allgemeinen die Strichstärke. Liegt der Leser nicht auf weißem Papier, liefert er ebenfalls „Schwarzpegel". In diesem Fall fällt der Vergleich des Indexregisterinhaltes mit 255 irgendwann positiv aus, und der Prozessor springt nach UEBRL und beginnt wieder bei START. Die sechs PLA-Befehle korrigieren den Stackpointer, da in eine andere Programmebene gesprungen wird.

Die Routine AUSW, die immer nach BITTST aufgerufen wird, stellt fest, ob ein gelesener Strich eine „0" oder

```
3E10                30 ;*********************************
3E10 86F6           31 READER STX XTEMP1
3E12 24FE           32        BIT PFFLAG          ;PUFFER LEER?
3E14 300E           33        BMI START           ;PUFFER FUELLEN
3E16 A6FF           34        LDX PNT
3E18 E4FA           35 EING   CPX BUF
3E1A F008           36        BEQ START
3E1C BDDD3E         37        LDA BUFFER,X
3E1F E6FF           38        INC PNT
3E21 A6F6           39        LDX XTEMP1
3E23 60             40        RTS
3E24                41 ;*********************************
3E24 204AFF         42 START  JSR IOSAVE
3E27 2C61C0         43        BIT PA
3E2A 30F8           44        BMI START           ;WARTEN AUF WEISS
3E2C 20CE3E         45        JSR SYNCH           ;SYNCH ZAEHLEN
3E2F 86F8           46        STX REF1
3E31 86F7           47        STX REF0
3E33 46F7           48        LSR REF0
3E35 20CE3E         49        JSR SYNCH           ;2. SYNCH
3E38 20803E         50        JSR AUSW
3E3B 907A           51        BCC FEHLER
3E3D 20BD3E         52        JSR BYTE
3E40 85FA           53        STA BUF             ;PUFFERENDE
3E42 A200           54        LDX #0
3E44 A900           55        LDA #0
3E46 85FB           56        STA CHCKL
3E48 85FC           57        STA CHCKH
3E4A 86FD           58 VOR    STX XTEMP
3E4C 20BD3E         59        JSR BYTE
3E4F A6FD           60        LDX XTEMP
3E51 48             61        PHA
3E52 0980           62        ORA #$80
3E54 9DDD3E         63        STA BUFFER,X
3E57 68             64        PLA
3E58 18             65        CLC
3E59 65FB           66        ADC CHCKL
3E5B 85FB           67        STA CHCKL
3E5D 9002           68        BCC NULL
3E5F E6FC           69        INC CHCKH
3E61 E8             70 NULL   INX
3E62 E4FA           71        CPX BUF
3E64 D0E4           72        BNE VOR
3E66 20BD3E         73        JSR BYTE
3E69 C5FB           74        CMP CHCKL
3E6B D04A           75        BNE FEHLER
3E6D 20BD3E         76        JSR BYTE
3E70 C5FC           77        CMP CHCKH
3E72 D043           78        BNE FEHLER
3E74 203FFF         79        JSR IOREST
3E77 A200           80        LDX #0
3E79 86FF           81        STX PNT             ;POINTER=0
3E7B 86FE           82        STX PFFLAG          ;PUFFER VOLL
3E7D 4C183E         83        JMP EING
3E80                84 ;*****************************
3E80                85 ;BIT AUSWERTEN
3E80 A5F7           86 AUSW   LDA REF0
3E82 4A             87        LSR
3E83 65F7           88        ADC REF0
3E85 85F9           89        STA REF             ;REF0+1/2
3E87 E4F9           90        CPX REF
3E89 1009           91        BPL EINS
3E8B 86F7           92        STX REF0            ;BIT IST NULL
3E8D A5F7           93        LDA REF0
3E8F 0A             94        ASL
3E90 85F8           95        STA REF1
3E92 18             96        CLC                 ;CARRY IST NULL
3E93 60             97        RTS
3E94 86F8           98 EINS   STX REF1
3E96 A5F8           99        LDA REF1
3E98 4A            100        LSR
3E99 85F7          101        STA REF0
3E9B 38            102        SEC                 ;CARRY IST EINS
3E9C 60            103        RTS                 ;BIT STEHT IN CARRY
3E9D               104 ;*****************************
3E9D 48            105 BITTST PHA
3E9E A200          106        LDX #0
3EA0 2C61C0        107 BIT    BIT PA
3EA3 10FB          108        BPL BIT
3EA5 E8            109 ZAEHL  INX
```

```
3EA6 A908     110          LDA #ZEIT
3EA8 E901     111  VERZ    SBC #1
3EAA D0FC     112          BNE VERZ
3EAC E0FF     113          CPX #255
3EAE F024     114          BEQ UEBRL
3EB0 2C61C0   115          BIT PA
3EB3 30F0     116          BMI ZAEHL
3EB5 68       117          PLA
3EB6 60       118          RTS
3EB7          119  ;*******************************
3EB7 20DDFB   120  FEHLER  JSR BELL1
3EBA 4C243E   121          JMP START
3EBD          122  ;*********************************
3EBD          123  ;BYTE EINLESEN
3EBD 48       124  BYTE    PHA
3EBE A008     125          LDY #8
3EC0 209D3E   126  NBIT    JSR BITTST
3EC3 20803E   127          JSR AUSW
3EC6 68       128          PLA
3EC7 6A       129          ROR
3EC8 48       130          PHA
3EC9 88       131          DEY
3ECA D0F4     132          BNE NBIT
3ECC 68       133          PLA
3ECD 60       134          RTS                    ;BYTE STEHT IN AKKU
3ECE          135  ;*******************************
3ECE 48       136  SYNCH   PHA
3ECF 209D3E   137          JSR BITTST
3ED2 68       138          PLA
3ED3 60       139          RTS
3ED4          140  ;*********************************
3ED4 68       141  UEBRL   PLA
3ED5 68       142          PLA
3ED6 68       143          PLA
3ED7 68       144          PLA
3ED8 68       145          PLA
3ED9 68       146          PLA
3EDA 4C243E   147          JMP START
3EDD 00       148  BUFFER  BRK                    ;PUFFERANFANG
             149          END
```

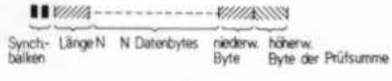
eine „1" repräsentiert. Ein Maß für die Strichstärke ist der Inhalt des Indexregisters X. Würde man ihn einfach mit einem absoluten Wert vergleichen und danach entscheiden, ob das Bit „0" oder „1" ist, dann müßte man den Leser mit genau definierter Geschwindigkeit über die Strichcodezeichen führen. Um dies zu vermeiden, benötigt man einen Referenzwert, der von der Geschwindigkeit abhängig ist. Er wird beim Überfahren der beiden breiten Sychronstriche am Anfang einer Zeile gewonnen (siehe Listing nach START). Im Programm ist er in Speicherzelle REF1 (Referenz für „1"-Strich) abgelegt. Teilt man diesen Wert durch zwei (Befehl LSR), dann erhält man den Referenzwert für einen „0"-Strich REF0.

Die Routine AUSW vergleicht jeden gelesenen Strich mit REF0 und REF1. Ist der X-Register-Inhalt größer als der arithmetische Mittelwert, dann handelt es sich um eine „1", ist er kleiner, handelt es sich um eine „0". Das Programm setzt abhängig davon das Carry-Bit oder löscht es (zur späteren Weiterverwendung in BYTE).

Wie ist der mc-Strichcode aufgebaut?

Jeder schwarze Strich stellt ein Bit dar: Ein breiter Strich bedeutet, daß dieses Bit „1" ist, ein schmaler, daß es „0" ist. Die Bits werden zu 8er-Gruppen (Bytes) zusammengefaßt. Links steht das niedrigstwertige Bit, rechts das höchstwertige. Das ASCII-Zeichen A (hexadezimal 41) sieht also folgendermaßen aus:

Ein schmaler Strich ist eine Einheit breit (ca. 0,3 mm); der Abstand zwischen zwei Strichen beträgt zwei Einheiten, und ein breiter Strich nimmt drei Einheiten ein.
Jede Strichcodezeile beginnt mit zwei breiten Synchronisationsbalken. Es folgt ein Byte, das die Anzahl der Datenbytes angibt. An die Datenbytes schließen sich niederwertiges und höherwertiges Byte einer 16-Bit-Prüfsumme an:



Die Prüfsumme wird durch Addition aller Datenbytes ermittelt. Grundsätzlich kann man auf diese Weise Programme in jeder Sprache veröffentlichen. Bei Basic-Programmen werden wir immer den vollen ASCII-Text abdrucken – also nicht den komprimierten Interncode (Tokens).

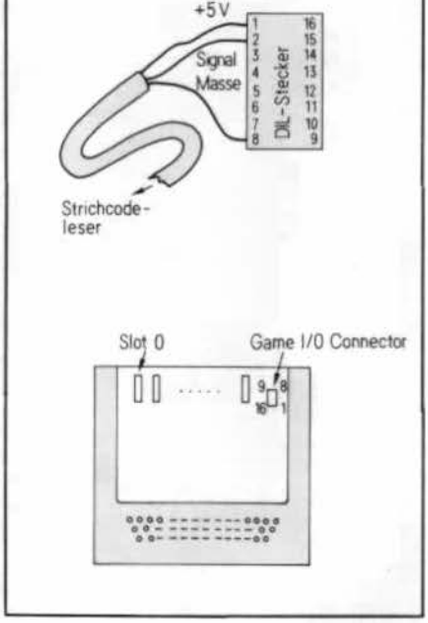


Bild 2. So wird der Strichcodeleser an den sogenannten „Game I/O Connector" des Apple (rechts hinten, in der Nähe des Video-Ausgangs) angeschlossen. Das Programm ist so ausgelegt, daß Bit 7 eines Ports als Eingangsleitung dient

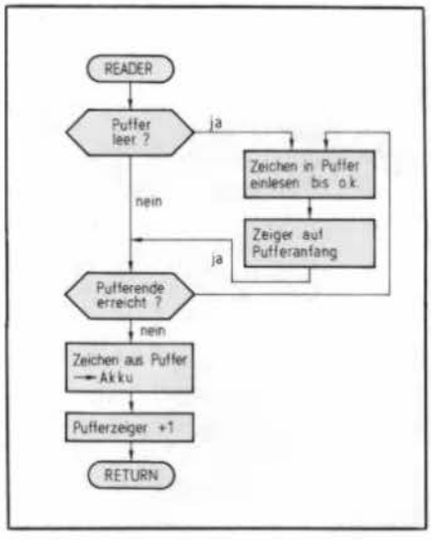
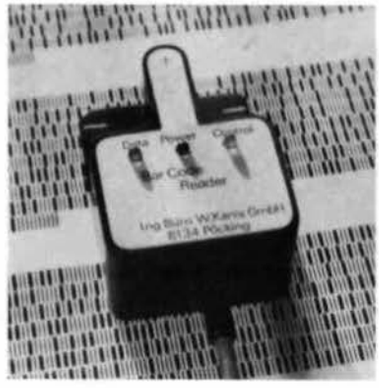


Bild 3. Das Programm READER übergibt dem aufrufenden Programm immer ein gültiges Zeichen im Akkumulator; es tritt beispielsweise an die Stelle der Tastaturabfrageroutine

Würden REF0 und REF1 während des gesamten Lesevorgangs gleich bleiben, dürfte sich auch die Geschwindigkeit nur unwesentlich ändern. Aus diesem Grund werden sie nach jedem gelesenen Bit aktualisiert. Das geht ganz einfach: War das Bit „0", ist das Zählergebnis (X) der neue Referenzwert REF0 (REF0 x 2 ist REF1). War das Bit „1", ist das Zählergebnis der neue Referenzwert REF1 (REF1/2 = REF0).
Die zulässige Geschwindigkeitsvariation wird dadurch so groß, daß man sie praktisch nur überschreiten kann, wenn man mit dem Leser innerhalb der Zeile stehenbleibt.

Lesestift für mc-Programme



Programme mittleren Umfangs aus Zeitschriften abzutippen ist nicht nur zeitraubend, sondern auch fehlerintensiv. Ein Kassettenversand macht zwar die Tipparbeit überflüssig, aber er ist mit Wartezeit und zusätzlichen Kosten verbunden. mc bietet deshalb ihren Lesern einen besonderen Service: Künftig werden regelmäßig interessante Programme im Strichcode abgedruckt. Dieser Code kann schnell und fehlerlos in den Computer eingegeben werden. Schnell, weil man die Zeilen nur mit einem „Stift" abzufahren braucht. Fehlerlos, weil eine Prüfsumme am Ende jeder Zeile dafür sorgt, daß nur richtig erkannte Zeichen übernommen werden.
Weil Strichcodeleser bisher keine billige Angelegenheit waren, beauftragte mc eine Firma mit der Entwicklung eines geeigneten Produkts. Dieser Leser ist in mehrfacher Hinsicht außergewöhnlich. Da er nur für flache Unterlagen gedacht ist, wurde er nicht als Stift, sondern als Element mit Auflagefläche ausgeführt (dieser Aufbau ist patentrechtlich geschützt). Dadurch ist gewährleistet, daß die Optik immer senkrecht zum Papier steht. Mit einem relativ preiswerten Abtastelement läßt sich auf diese Weise eine genügend hohe Auflösung erzielen. Nur so konnte ein Preis realisiert werden, der wesentlich niedriger ist als der aller vergleichbaren Produkte auf dem Markt (135 DM inkl. MwSt. und Nachnahme-Versand bei: Ing.-Büro W. Kanis GmbH, Lindenberg 113, 8134 Pöckung). Der Leser wird mit 5 V versorgt und liefert ein Signal (schwarz/weiß) im TTL-Pegel. D. h., er kann an jeden Computer angeschlossen werden, der mindestens eine frei programmierbare Eingangsleitung hat. Ist kein sogenannter User-Port vorhanden, kann auch das Kassetten-Interface zweckentfremdet werden. Geeignete Einleseprogramme wird mc für die verbreiteten Computermodelle veröffentlichen.

Der Volkscomputer kommt

Einen volltönenden Namen hat sich die Firma Commodore für ihren „Consumer-Computer" VC-20 in Deutschland ausgedacht: Volkscomputer! mc bringt hier die wichtigsten technischen Daten:
Speicherkapazität: 5-KByte-RAM, bis 32 KByte ausbaubar.
Befehlsvorrat des Commodore-Basic:

CLOSE	CLR	CMD	CONT	DATA
DEF FN	DIM	END	FOR..NEXT	GET
GOSUB..RETURN	GOTO	IF..THEN		INPUT
INPUT	LET	LIST		LOAD
NEW	ON..GOSUB	OPEN		POKE
PRINT	READ	REM		RESTORE
RUN	SAVE	STOP		SYS
VERIFY	WAIT	ABS		ASC
ATN	CHR	COS		EXP
FRE	INT	LEFT$		LEN
LOG	MID$	PEEK		POS
RIGHT$	RDN	SGN		SIN
SPC	SQR	STATUS		STR$
TAB	TAN	TIMT		TIME$
USR				

Farben:	Insgesamt 24 Farben – davon 8 für Zeichen und Bildschirmumrandung, 16 für Bildschirmhintergrund. Programmiertasten für Grundfarben schwarz, weiß, rot, blau, grün, gelb, purpur, zyan
Sound:	3 Ton-Generator für Musik. 1 Ton-Generator für Sprache und Geräuscheffekte. Jeder Generator verfügt über 3 Oktaven. Zur Wiedergabe werden die Fernsehlautsprecher genutzt.
Zeichen/Zeile:	Je 22 Zeichen in 23 Zeilen. 64 ASCII-Zeichen, grafische Zeichen, Groß- und Kleinschreibung über Umschaltung.

Luidger Röckrath

Der geknackte TRS-80

Über den internen Aufbau von Basic-Interpretern legen die Hersteller verbreiteter Tischcomputer meist den Mantel des Schweigens. So erfordert es schon einige Arbeit, diese Nuß zu knacken. Die Ergebnisse dieser Arbeit (über 500 Adressen und einiges mehr) sind im folgenden dargestellt.

Im Franzis-Sonderheft „Mikrocomputer-Anwendungen" erschienen für mehrere 6502-Computer RAM/ROM-Listings [1], d. h. Aufstellungen aller wichtigen Adressen sowohl für Programmspeicher als auch für Datenspeicher. Dies ist erforderlich, weil die meisten Hersteller sich über diese Dinge ausschweigen, obwohl sie zur optimalen Ausnutzung des Computers für den Anwender aus verschiedenen Gründen notwendig sind [2].
Erstens für den Basic-Programmierer, der Programme für andere Computer adaptieren muß und dazu z. B. die Tastaturadressen benötigt, aber auch für Programmierer, die es entgegen allen Hindernissen nicht sein lassen können, ihren Computer in Maschinensprache zu programmieren, sind besonders die Adressen der Ein-/Ausgaberoutinen von Interesse. Und schließlich für jene Unverbesserlichen, die meinen, ihr Basic sei noch nicht gut genug und könnte durch einige zusätzliche Befehle sinnreich erweitert werden.
Das RAM/ROM-Listing für den TRS-80 enthält für alle drei Gruppen etwas: für den Basic-Programmierer alle wichtigen I/O-Adressen (einschließlich aller Tastaturadressen), für die passionierten Maschinenprogrammierer alle wichtigen Unterprogramme, die sie zum Anpassen fremder oder zum Entwickeln eigener Programme benötigen. Um es dieser Gruppe besonders leicht zu machen, enthält die Tabelle eine genaue Beschreibung, was das Unterprogramm eigentlich tut und welche Register verändert werden. Die letzte Gruppe sei auf den vorletzten Abschnitt dieses Artikels verwiesen, in dem ausführlich die zahlreichen Möglichkeiten des Eingriffs in den Basic-Interpreter erläutert sind.

Einige grundsätzliche Bemerkungen

Wer schon mal in der Tabelle gestöbert hat, der wird sich gewundert haben, daß nur die Beeinflussung der 8080-Register angegeben ist, obwohl der TRS-80 einen Z80 als Zentraleinheit beinhaltet, der doch erheblich mehr Register hat. Des Rätsels Lösung ist ebenso einfach wie enttäuschend: Der TRS-80 arbeitet mit einem 8080-Basic. Man hat sich nur die Mühe gemacht, die Sprünge, wenn möglich, in relative Sprünge umzuwandeln. Lediglich die systemspezifische Ein-/Ausgabe ist im Z80-Code programmiert; dort wird auch einmal das X-Indexregister benutzt, aber zuvor gerettet. So stehen der ganze Zweitregistersatz und die Indexregister für Erweiterungen zur Verfügung.
Das Level-2-Basic des TRS-80 ist ein Microsoft-Basic, d. h. der Programmtext wird nicht so abgespeichert wie er eingegeben wurde, sondern zuvor in einen Zwischencode (1 Byte pro Befehl, zwischen hex 80 und FA) umgewandelt. Diese Zwischencodes sind in [3] auf Seite E/1 aufgelistet. Bei der Programmausführung muß jetzt also nicht mehr das ganze Befehlswort erkannt und dann aus einer Tabelle eine Adresse herausgesucht werden, sondern der Zwischencode dient als Zeiger auf eine Tabelle, die die Ansprungadressen der Befehle und Funktionen enthält. So nimmt die Interpretation des Programmes nicht mehr so viel Zeit in Anspruch wie z. B. beim Level-1-Basic. Nach der gleichen Methode arbeiten die meisten handelsüblichen Interpreter (PET, Apple usw.).

Unterschiedliche Versionen

Das RAM/ROM-Listing wurde von einer Version (V.2) des Level-2-Basic erstellt, die Mitte 1980 verkauft wurde. Bei dieser Version sind der READ/DATA-Fehler [4] und das Prellen der Eingabetastatur ausgemerzt. Zum Vergleich wurde eine ältere Version (V.1) herangezogen, welche die beiden oben genannten Fehler noch hat. Die Unterschiede der V.2 gegenüber der V.1 seien im folgenden dargestellt:

1. Der Tastendruck SHIFT ← wird nicht mehr erkannt aufgrund einer Änderung in der Tastaturdecodierungstabelle. (SHIFT ← hat nun die Funktion einer CONTROL-Taste.)
2. Zugunsten der Entprellung wurden die Texte „RADIO SHACK LEVEL 2 BASIC" und „MEMORY SIZE" zu „R/S L2 BASIC" und „MEM SIZE" gekürzt. Dies hat noch eine Änderung der Zeiger zur Folge.
3. Um beim Laden von der Kassette die Fehlerträchtigkeit zu erniedrigen, wurden zwei Schleifen geringfügig verlängert.
4. Einige kleine Änderungen am PRINT-Befehl, die bewirken, daß

a) mehrere @ in einer PRINT-Anweisung möglich sind,
b) nicht mehr zwischen @ und SHIFT @ unterschieden wird,
c) das Argument von TAB nicht mehr modulo 64, sondern modulo 128 vorbehandelt wird.

5. Der READ/DATA-Fehler wurde behoben.
6. Durch eine Änderung bei CLOAD ist es nicht mehr möglich, eine Kassettenrecorder-Nummer anzugeben (CLOAD#-1, „A" führt zu Fehlermeldung).

Wie von Radio Shack bestätigt, existieren auch andere Versionen, bei denen z. B. nur der READ-/Data-Fehler ausgemerzt ist. Bei der Programmentwicklung sollten diese Unterschiede beach-

tet werden, um die Kompatibilität zu wahren.

Eingriff in den Interpreter

Der Level-2-Interpreter ist zwar in zwei ROMs abgespeichert, und somit nicht zu ändern, aber an sehr vielen Stellen verzweigt die Programmausführung ins RAM. Dort kann leicht in den Programmablauf eingegriffen werden.

1. Alle wichtigen Ein-/Ausgaberoutinen werden über die DCBs angesprungen, die die Treiberadressen enthalten. Ändert man diese Adressen, so können leicht eigene Ein-/Ausgaberoutinen mit dem Basic-Interpreter in Verbindung treten [5], [6], [7].

2. Wichtige Unterprogramme des Basic werden über die RSTs 8 bis 20 (hex) angesprungen, deren Vektoren im RAM liegen. In praktisch allen Programmsegmenten des Basic-Interpreters werden diese Unterprogramme benötigt. Eine Analyse des entsprechenden Programmteiles zeigt schnell, wie man zweckmäßig eingreifen kann. Dann wird nur noch der Vektor geändert, nach Ansprung des entsprechenden RST geprüft, ob das richtige Programmsegment aufruft und schon kann die eigene Erweiterung folgen [8].

3. Eine einfache und fast unerschöpfliche Möglichkeit des Eingriffes sind die zahlreichen Disk-Basic-Befehle und Funktionen. Diese werden erkannt; sodann wird ins RAM verzweigt, wo Sprünge zu den eigentlichen Programmsegmenten zur Ausführung dieser Befehle und Funktionen stehen. Diese Sprünge zu ändern und eigene neue Befehle zu implentieren, ist kein Problem. Wie [9] zeigt, muß dies noch nicht einmal dazu führen, daß die Zusammenarbeit mit dem Disk-Basic gestört ist.

4. Zur Erweiterung vorhandener Befehle des Level-2-Basic durch das Disk-Basic (z. B. PRINT#, was im Disk-Basic zum Schreiben auf Diskette dient), rufen diese Befehle Unterprogramme im RAM auf. Normalerweise sind diese Adressen mit Return-Befehlen (C9) belegt, aber sie können leicht durch entsprechende Sprünge ersetzt werden. Auch hier analysiert man zweckmäßigerweise das zu erweiternde Programmsegment und wird bald herausfinden, wie man günstig eingreifen kann. Man muß beachten, daß eine Adresse unter Umständen aus verschiedenen Programmteilen aufgerufen wird.
Nun schnell den TRS-80 angeschaltet und an die Arbeit! Wem das hier alles noch zu abstrakt ist, der möge sich einmal [7], [8] und [9] anschauen, drei Beispiele für die ersten drei Methoden. Sollte es jemandem an Ideen ermangeln, auch dem kann geholfen werden. [10] und [11] dürften für die nächsten Feierabende ausreichen.

Das RAM/ROM-Listing

Das RAM/ROM-Listing enthält über 500 ROM-, RAM- und I/O-Adressen. Bei wichtigen Unterprogrammen ist angegeben, welche Register verändert werden (xx). Ist in dieser Spalte ein konkreter Zahlenwert angegeben, so bedeutet dies, daß nach der Rückkehr von diesem Unterprogramm das entsprechende Register immer diesen Wert beinhaltet. Das Register HL dient bei der Ausführung von Basic-Programmen als Zeiger auf den Programmtext. Hat es diese Funktionen, so ist ein P in der entsprechenden Spalte zu sehen.

Literatur

[1] *Martin, R.; Smode, D.:* ROM und RAM in PET und CBM. Franzis-Sonderheft „Mikrocomputer-Anwendungen".

[2] *Feichtinger, H.:* Meine Meinung... FUNKSCHAU 1979, Heft 24, Seite 1418.

[3] Radio Shack: Level II Basic Reference Manual.

[4] *Neske, W.:* TRS-80 – ein Computer mit kleinem Fehler. Franzis-Sonderheft „Programme für Kleincomputer und Taschenrechner".

[5] *Duddek, G.:* Ein preiswerter Drucker für den TRS-80. Franzis-Sonderheft „Hobbycomputer 2".

[6] *Hofer, R.:* TRS-80-Treiberprogramm für Schreibmaschinen-Drucker. FUNKSCHAU 1980, Heft 7, Seite 91.

[7] *Röckrath, L.:* Repeatfunktion für den TRS-80 (in Vorbereitung).

[8] *Röckrath, L.:* RESTORE N für den TRS-80 (in Vorbereitung).

[9] *Röckrath, L.:* Double-Precision-Funktionen (in Vorbereitung).

[10] *Feichtinger, H.:* Super-Basic für AIM-65 und PC-100. FUNKSCHAU 1980, Heft 25, Seite 103.

[11] *Klein, R. D.:* S-100-System mit Basic und schnellem Kassetten-Interface. Franzis-Sonderheft „Hobbycomputer 1".

ROM- und RAM-Adressen

ROM-Adressen Level-II-Basic-Interpeter

Initialisierung und Ein-/Ausgabe					AF	BC	DE	HL
0000	0000	00000	00000	RST 0, Basic-Kaltstart				
0008	0008	00008	00008	RST 18, Basic-UP s. 1C96				
0010	0010	00016	00016	RST 10, Basic-UP s. 1D78				
0018	0018	00024	00024	RST 18, Basic-UP s. 1C90				
0020	0020	00032	00032	RST 20, Basic-UP s. 25D9				
0028	0028	00040	00040	RST 28, wird bei Drücken der Break-Taste angespr.				
002B	002B	00043	00043	INCH1-Ansprung (über DCB), Tastaturabfrage: ASCII-Code gedrückter Taste in A, wenn keine Taste gedrückt ist, A=0	xxxx		4015	
0030	0030	00048	00048	RST 30, unbenutzt				
0033	0033	00051	00051	OUTCH-Ansprung (über DCB), Ausgabe des Akkuinhaltes auf den Bildschirm	xx		401D	
0038	0038	00056	00056	RST 38H, unbenutzt				
003B	003B	00059	00059	PRINT-Ansprung (über DCB), Ausgabe des Akkuinhaltes auf den Drucker	xxxx		4025	
0049	004F	00073	00079	INCH2, wie INCH1, es wird aber gewartet, bis eine Taste gedrückt wird	xxxx		4015	

					AF	BC	DE	HL
0050	005F	00080	00095	Tabelle für die Tastaturdecodierung				
0060	0065	00096	00101	DELAY, Zeitschleife 14,6 µs × BC	0044	0000		
0066	0074	00102	00116	Reset-Ansprung (NMI-Vektor).				
				Wenn Floppy angeschlossen, Basic-Kaltstart, sonst Warmstart				
0075	0104	00117	00260	Basic-Initialisierung:				
0075	008D	00117	00141	DCBs, RST-Vektoren und I/O-Buffer einrichten				
008E	00AB	00142	00171	Disk-Basic-Zeiger initialisieren				
				(werden beim Laden des Disk-Basic geändert)				
00AC	00F8	00172	00248	MEM SIZE anfordern oder selbst Speicherende suchen				
00F9	0104	00249	00260	NEW, „R/S L2 BASIC" ausdrucken und Sprung zur Hauptschleife				
0105	010D	00261	00269	Text „MEM SIZE"				
010E	011B	00270	00283	Text „R/S L2 BASIC(CR)"				
011C	012C	00284	00300	Tastaturentprellung				
012D	0131	00301	00305	L3-Error				
0132	0132	00306	00306	POINT A=00 } Ansprung der drei Grafikbefehle				
0135	0135	00309	00309	SET A=80 } und Setzen des Flags (A).				
0138	0138	00312	00312	RESET A=01 }				
013A	017D	00314	00381	Aus den Koordinaten werden die Adresse im Bildschirmram				
				und die Maske errechnet				
017E	019C	00382	00412	In Abhängigkeit von dem Flag wird ein POINT,				
				SET oder RESET ausgeführt				
019D	01C8	00413	00456	INKE$-Funktion				
01BC	01C8	00444	00456	Leerstring nach X				
01C9	01D2	00457	00466	CLS-Befehl, Bildschirm wird gelöscht	1Fxx			
01D3	01D8	00467	00472	RANDOM-Befehl				
				(als zufällige Größe wird das R-Register benutzt)				
01D9	01F7	00473	00503	Impuls auf Kassette ausgeben	xxxx	00		FC00
01F8	01FD	00504	00509	Kassettenrecorder abschalten	xxxx			
01FE	021D	00510	00541	Kassettenrecordernummer decodieren	xxxx	xxxx	xxxx	P
				und Kassettenrecorder einschalten				
0215	021D	00533	00541	Kassettenrecorder einschalten	xxxx			
021E	0220	00542	00544	Bit 7 von Port (FF) zurücksetzen	xxxx			FC00
0221	022B	00545	00555	(403D) ∧ H ∨ L nach (403D) und zum Port FF	xxxx			
022C	0234	00556	00564	Stern in rechter, oberer Ecke des Bildschirms umschalten	xxxx			
0235	0260	00565	00608	Byte von Kassette lesen (wird im Akku übergeben)	xxxx			
0241	0260	00577	00608	Bit (b) von Kassette lesen (2 . A + b ⧫ A)	xxxx			FC00
0261	0283	00609	00643	Byte (im Akku) zweimal auf Kassette aufzeichnen				
0264	0283	00612	00643	Byte (im Akku) auf Kassette aufzeichnen				
0284	0292	00644	00658	Kassette einschalten, 255 mal 0 und A5 aufzeichnen	xxxx	xxxx	xxxx	P
0293	02A8	00659	00680	Kassette einschalten, auf A5 warten und Sternchen	2Axx	xxxx	xxxx	P
				auf Bildschirm setzen				
02A9	0329	00681	00809	SYSTEM-Befehl				
02B2	02B2	00690	00690	Ansprung				
02CE	0313	00718	00787	Objektfile von Kassette laden				
0314	031C	00788	00796	Wort (in HL) von Kassette laden (L, H)	xxxx			xxxx
031D	0329	00797	00809	Ansprung des Objektfile oder irgendeiner anderen Adresse				
032A	0347	00810	00839	Ausgabe des Akkuinhaltes auf Bildschirm (00),	xx			
				Drucker (01) oder Kassette (80) in Abhängigkeit von (409C)				
033A	0347	00826	00839	OUTCH2, wie OUTCH, zusätzlich Cursorposition nach (40A6)	xx			
0348	0357	00840	00855	Position des Cursors in der Bildschirmzeile nach A	xxxx			
0358	0360	00856	00864	INCH3, wie INCH1	xxxx			
0361	0383	00865	00899	INLINE, Eingabe von max. 240 Zeichen in den I/O-Buffer	xxxx		401D	xxxx
				mit allen Cursorfunktionen. Wenn Breack gedrückt wird,				
				Rückkehr mit gesetzten Carry-Flag, bei Drücken von Clear				
				wird der Bildschirm gelöscht und die Eingabe beginnt				
				von vorne. HL enthält die Bufferanfangsadresse –1				
0384	038A	00900	00906	INCH4, wie INCH2	xxxx			
038B	039B	00907	00923	CR auf Drucker ausgeben, wenn Druckkopf nicht	xxxx			
				in Position 0, (409C) auf 0 setzen				
039C	03C1	00924	00961	Zeichen in A auf Drucker ausgeben, Zeichenzähler (409B)				
				incrementieren und bei 0A, 0C und oD auf 0 setzen				
03C2	03E2	00962	00994	Routine zum Aufruf der Ein-/Ausgabe über die DCBs.				
				(DCB-Typ in B, DCB-Adresse in DE)				
03E3	0457	00995	01111	Tastaturabfrage und Decodierung				
				(Ansprung nur über INCH1-4, da nur dann die				
				Register gerettet werden, siehe dort)				
0458	058C	01112	01420	Bildschirmausgabe				
				(Ansprung nur über DCB, da nur dann Register gerettet werden				
				und die Cursoradresse übergeben wird, siehe OUTCH)				
04B8	058C	01208	01420	Bildschirm-Steuerbefehle				

Der vielseitige Weg für zukünftige Mikrocomputer-Entwicklungen.

Die Vielfalt der Mikrocomputer stellt hohe Anforderungen an die Systeme zum Mikrocomputer-Schaltungsentwurf: flexibel, modular, wirtschaftlich, zukunftsorientiert.
Die modulare MDL-Serie 8500 bietet diese Eigenschaften, die Ihre Investitionen so gering als möglich halten. Die neuen Systeme, Serie 8500, bieten vielseitige Einsatzmöglichkeiten mit „In-circuit"- und Echtzeit-Emulation, für 26 verschiedene 8 Bit- und 16 Bit-Mikrocomputer.

Single user-Systeme und Multi user-Systeme können gemäß den Ansprüchen in ihren Entwicklungslabors konzipiert werden. Integrationsstationen für Ihren Host-Computer erlauben die Software-Entwicklung mit den Methoden, die der Aufgabenstellung gerecht werden.

Schulung, Training, Service
Mikrocomputer-Entwicklungen gehören zum Erfahrungsschatz von Tektronix. Neben einem umfangreichen Schulungsprogramm für den wirtschaftlichen Einsatz Ihres Mikrocomputer-Entwicklungssystem, gibt es ausgefeilte Trainingsprogramme über Mikrocomputer und die Benutzung von Entwicklungssystemen. Außerdem bietet Ihnen Tektronix einen umfassenden Service. Informieren Sie sich über das Tektronix-System, über unser Schulungs- und Serviceangebot.

452.c.381.e

Single user-System 8550

Multi user-System 8560

Integrationsstation 8540

Tektronix GmbH
Sedanstraße 13–17, 5000 Köln 1
Tel. (02 21) 77 22-1, Tx. 8 885 417

Geschäftsstellen:
Ernst-Reuter-Platz 3–5, 1000 Berlin 10 (West)
Tel. (0 30) 313 90 81–83, Tx. 1 85 804

Große Bergstraße 213, 2000 Hamburg 50
Tel. (0 40) 38 01 91, Tx. 213 749

Schönhauser Straße 62, 5000 Köln 51
Tel. (02 21) 37 50 81–88, Tx. 8 885 541

Kriegsstraße 39, 7500 Karlsruhe 1
Tel. (07 21) 2 79 81, Tx. 7 825 301

Ehrenbreitsteiner Str. 36, 8000 München 50
Tel. (0 89) 14 85-1, Tx. 5 22 953

Donaustraße 36, 8500 Nürnberg 60
Tel. (0911) 64 60 81, Tx. 6 26 255

Tektronix®
COMMITTED TO EXCELLENCE

					AF	BC	DE	HL
04B8	04BC	01208	01212	Cursor ON	(0E)			
04BD	04BF	01213	01215	Cursor OFF	(0F)			
04C0	04CD	01216	01229	Cursor HOME (64 cpl)	(1C)			
04CE	04D9	01230	01241	Cursor BACKSPACE	(08)			
04DA	04EB	01242	01259	Cursor ↓	(18)			
04E7	04EB	01255	01259	Cursor ←	(1A)			
04EC	04F5	01260	01269	Cursor ↑	(19)			
04F1	04F5	01265	01269	Cursor →	(1B)			
04F6	0505	01270	01285	32cpl	(17)			
0506	0540	01286	01344	Auswertung der Steuerbefehle				
0541	0563	01345	01379	Zeichen auf Bildschirm setzen und Scroll, wenn nötig				
0564	0572	01380	01394	New-Line	(0A-0D)			
0573	058C	01395	01420	Clear to end of line	(1E)			
057C	058C	01404	01420	Clear to end of frame	(1F)			
058D	05D0	01421	01488	Drucker-Treiber				
05D1	05D8	01489	01496	Drucker bereit?	xxxx			
				j,Z=1.				
05D9	0673	01497	01651	Unterprogramm für INLINE wie INLINE, maximale Anzahl	xxxx	xxxx	401D	
				der Zeichen in B, Bufferanfangsadresse in HL				
				Bei Rückkehr Anzahl der eingegebenen Zeichen in B				
0674	06CB	01652	01739	Wenn Breack gedrückt ist, Sprung zur Basic-				
				Initialisierung, sonst Testen ob Floppy vorhanden				
				Wenn ja, DOS-Laden und starten				
06CC	06CC	01740	01740	Adresse für Rücksprung von SYSTEM-Programmen zum Basic				
				(Basic-Warmstart)				
06D2	0707	01746	01799	RST-Vektoren, DCBs usw. (werden bei Initialisierung				
				ins RAM übertragen (4000–4035)				
Arithmetik								
0708	07F7	01800	02039	X + 1 ↓ X	xxxx	xxxx	xxxx	xxxx
0710	07F7	01808	02039	(HL...) + X ↓ X	xxxx	xxxx	xxxx	xxxx
0713	07F7	01811	02039	BCDE − X ↓ X	xxxx	xxxx	xxxx	xxxx
0716	07F7	01814	02039	BCDE + X ↓ X	xxxx	xxxx	xxxx	xxxx
0778	077C	01912	01916	0 ↓ X	0044			
07A8	07B1	01960	01969	Rundung				
07B2	07B6	01970	01974	OV-Error				
07B7	07C2	01975	01986	Festkommaaddition:	xxxx	xx	xxxx	+2
				(HL...) + CDE ↓ CDE				
07D7	07F7	02007	02039	CDE um A-Bitpositionen	00xx	xxxx	xxxx	xx
				nach rechts schieben				
07F8	07FB	02040	02043	Konstante 1!				
07FC	0808	02044	02056	Konstanten für LOG-Reihe				
0809	0896	02057	02189	LOG(X) ↓ X	xxxx	xxxx	xxxx	xxxx
0841	0896	02057	02189	X . ln2 ↓ X	xxxx	xxxx	xxxx	xxxx
0847	0896	02119	02189	X . BCDE ↓ X	xxxx	xxxx	xxxx	xxxx
0897	0906	02199	02311	X / 10 ↓ X	xxxx	xxxx	xxxx	xxxx
08A2	0906	02211	02310	BCDE / X ↓ X	xxxx	xxxx	xxxx	xxxx
0907	093D	02311	02365	Vorbehandlung der Exponenten und Vorzeichen				
				bei Multiplikation und Division				
093E	0954	02366	02388	X . 10 ↓ X	xxxx	xxxx	xxxx	xxxx
0955	0963	02389	02403	Text X (für Single und Double Presision)	xxxx			
				Wenn X = 0, Z,P				
				Wenn X < 0, C,S				
				Wenn X > 0, keine				
0964	0976	00404	02422	Akku in Fließkommazahl (in X) umformen	xxxx	xxxx	xxxx	xxxx
0977	0989	02423	02441	ABS(X) ↓ X	xxxx	xxxx	xxxx	xxxx
0982	0989	02434	02441	− X ↓ X	xxxx			4123
098A	0993	02442	02451	SGN(X) ↓ X	xxxx			xxxx
0994	09A3	02452	02467	Test X, auch für Integer (siehe oben 0955)	xxxx			xxxx
				bei String in X TM-Error				
09A4	09B0	02468	02480	X ↓ (SP)			xxxx	
09B1	09BE	02481	02494	(HL...) ↓ BCDE ↓ X		xxxx	xxxx	+4
09B4	09BE	02484	02494	BCDE ↓ X			xxxx	
09BF	09CA	02495	02506	X ↓ BCDE		xxxx	xxxx	4125
09C2	09CA	02498	02506	(HL...) ↓ BCDE		xxxx	xxxx	+4
09CB	09DE	02507	02526	X ↓ (HL..)	xxxx	00	4125	+4
09D2	09DE	02514	02526	(HL...HL+(40AF)) ↓ (DE..)	xxxx	00	+4	+4

Fortsetzung folgt

Dipl.-Phys. Peter Kittel

Große Sprünge mit dem 6502

Sollen mehrere Maschinenprogramme (Moduln) zusammengefügt werden, ist es von Vorteil, wenn sie im Speicher frei verschiebbar sind. Sonst müssen nämlich bei einer Umordnung der Programme alle absoluten Adressen mitgeändert werden, was meist sehr mühsam ist.

Um ein Programm frei verschiebbar zu machen, dürfen statt absoluter Adressen nur noch relative (zur derzeitigen Programmadresse) verwendet werden. Im 6502-Befehlssatz ist das aber leider nur bei den kurzreichweitigen (±128 Bytes) Branch-Befehlen möglich. Mit den weiter unten beschriebenen Hilfsroutinen AKTADR und VERSATZ werden dem Benutzer jetzt aber drei neue Pseudo-Befehle zur Verfügung gestellt, die 8 bzw. 5 Bytes lang sind. Mit ihnen lassen sich völlig frei verschiebbare Programme verwirklichen:

1. JMP relativ:
JSR AKTADR
VL ←
VH
JMP VERSATZ
2. JSR relativ:
JSR AKTADR
VL ←
VH
JSR VERSATZ
3. Ermittle derzeitige Programmadresse:
JSR AKTADR
NOP
NOP ←

Die Hilfsroutinen

Das Unterprogramm AKTADR ermittelt die aktuelle Programmadresse, erhöht sie um 2 (um die Versatzbytes einbauen zu können, s. u.) und speichert sie als 16 Bit langen Pointer in der Zero Page ab. Dieser Pointer zeigt jeweils auf das mit einem Pfeil gekennzeichnete Byte. Die Routine benutzt ein ähnliches Verfahren, wie es in [1] und [2] vorgestellt wurde.

VL und VH bilden zusammen einen 16 Bit großen Versatz, der in der Routine VERSATZ zum Pointer addiert wird. Der Versatz ist ab dem gekennzeichneten Byte zu zählen. Da der Versatz 16 Bit groß ist, kann man mit den Sprüngen *jeden* anderen Befehl im 64-KByte-Speicherbereich erreichen, also auch rückwärts springen.

Der dritte Pseudobefehl von 5 Byte Länge (wobei statt der NOP-Befehle auch beliebig anderes stehen kann) wird gebraucht, um ins Programm eingebaute Tabellen ansprechen zu können (s. u., Ladeprogramm).

Bild 1 zeigt ein Disassembler-Listing der beiden Routinen. Beide sind etwas länger als unbedingt nötig, weil mit einigem Aufwand *alle* Registerinhalte gerettet werden – in VERSATZ auf den Stack, in AKTADR in zwei Zero Page-Speicherstellen. Zusätzlich ist in AKTADR noch ein PET-2001-spezifischer Teil zur Abfrage der Break-Taste eingebaut. So hat man mehr Sicherheit gegen Maschinenprogramme mit „unendlichen" Schleifen. Dieser Programmteil be-

```
949   03B5  08          PHP IMPLIED
950   03B6  85 00       STA ZER PAG
952   03B8  68          PLA IMPLIED
953   03B9  85 0C       STA ZER PAG
955   03BB  18          CLC IMPLIED
956   03BC  68          PLA IMPLIED
957   03BD  69 02       ADC IMMED.
959   03BF  85 01       STA ZER PAG
961   03C1  68          PLA IMPLIED
962   03C2  69 00       ADC IMMED.
964   03C4  85 02       STA ZER PAG
966   03C6  48          PHA IMPLIED
967   03C7  A5 01       LDA ZER PAG
969   03C9  48          PHA IMPLIED
970   03CA  AD 03 02    LDA ABSOLUT
973   03CD  C9 04       CMP IMMED.
975   03CF  D0 03       BNE REL 03D4
977   03D1  4C BB C5    JMP ABSOLUT
980   03D4  A5 0C       LDA ZER PAG
982   03D6  48          PHA IMPLIED
983   03D7  A5 00       LDA ZER PAG
985   03D9  28          PLP IMPLIED
986   03DA  60          RTS IMPLIED

987   03DB  08          PHP IMPLIED
988   03DC  48          PHA IMPLIED
989   03DD  98          TYA IMPLIED
990   03DE  48          PHA IMPLIED
991   03DF  A5 01       LDA ZER PAG
993   03E1  D0 02       BNE REL 03E5
995   03E3  C6 02       DEC ZER PAG
997   03E5  C6 01       DEC ZER PAG
999   03E7  A0 00       LDY IMMED.
1001  03E9  18          CLC IMPLIED
1002  03EA  B1 01       LDA (IND),Y
1004  03EC  65 01       ADC ZER PAG
1006  03EE  48          PHA IMPLIED
1007  03EF  C8          INY IMPLIED
1008  03F0  B1 01       LDA (IND),Y
1010  03F2  65 02       ADC ZER PAG
1012  03F4  85 02       STA ZER PAG
1014  03F6  68          PLA IMPLIED
1015  03F7  85 01       STA ZER PAG
1017  03F9  68          PLA IMPLIED
1018  03FA  A8          TAY IMPLIED
1019  03FB  68          PLA IMPLIED
1020  03FC  28          PLP IMPLIED
1021  03FD  6C 01 00    JMP ABS IND
```

Bild 1. Die Hilfsroutinen AKTADR (03B5-03DA) und VERSATZ (03DB-03FF) ermitteln die aktuelle Adresse des Programms und den relativen Versatz bei Sprung- und Ladebefehlen des Prozessors 6502

steht aus den Speicherstellen 03CA-03D3 und kann auf anderen Rechnern ganz entfallen. Bei CBMs mit neuen ROMs müssen auch andere Adressen eingesetzt werden.
Die beiden Routinen stehen im oberen Teil des Tape Buffers 2 im PET. Sie sind aber selbst schon frei verschiebbar, so daß sie auf anderen Rechnern auch an beliebig anderer Stelle untergebracht werden können. Man muß sich allerdings für eine bestimmte, unveränderliche Stelle entscheiden, um die Routinen jederzeit mit dem normalen, absolut adressierten JSR-Befehl aufrufen zu können.

So arbeiten AKTADR und VERSATZ

AKTADR rettet also zunächst Akku und Statusregister in zwei Speicherstellen, holt dann die vom JSR-Befehl dorthin gebrachte Rücksprungadresse vom Stack, erhöht sie um 2, speichert sie als Pointer in die Zero Page und schiebt sie wieder auf den Stack zurück. Bei gedrückter Break-Taste wird zum Warm Start of Basic (READY-Meldung beim CBM) gesprungen. Sonst werden die Register wieder auf den alten Stand gebracht und per RTS ins Hauptprogramm zurückgekehrt. Dabei werden wegen der Erhöhung der Rücksprungadresse die 2 Bytes hinter dem JSR-Befehl übersprungen, so daß in ihnen auf einfache Weise der 16 Bit lange Versatz für die relative Adressierung übergeben werden kann.
VERSATZ rettet die Register auf den Stack und erniedrigt den Pointer gleich wieder um 1, um auch an das erste (niederwertigere) Versatzbyte heranzukommen. Die beiden Versatzbytes werden mit der indirekt-indizierten Adressierung geholt und zum Pointer hinzuaddiert. Daraufhin werden die alten Registerinhalte wiederhergestellt. Als Abschluß erfolgt kein Return- sondern ein absolut-indirekter Sprungbefehl, der sich sein Sprungziel vom umgerechneten Pointer holt. VERSATZ ist also wegen des Sprunges am Schluß eigentlich keine Subroutine, sondern ein Vorspann vor dem angesprungenen Programmteil, so daß sie gleichermaßen für Sprünge wie für Subroutinen-Aufrufe zu gebrauchen ist.

Ladeprogramm und Beispiele

Bild 2 zeigt ein (natürlich frei verschiebbares) *Ladeprogramm*, das als Vorspann eines im oberen Teil des RAM liegenden Maschinenprogramms die beiden Routinen an eine *feste* Stelle speichert. Dazu wird zunächst eine Einfachversion von AKTADR „zu Fuß" erzeugt (*Bild 3*) und gleich aufgerufen, um wieder mit der indirekt-indizierten Adressierung die komplett ausgebauten Routinen aus einer dahinter folgenden Tabelle (1C40-1C8A) zu holen und umzuspeichern. Der Programmabschnitt 1C2E-1C3F zeigt, wie mit AKTADR umzugehen ist, aber Vorsicht: hier wurde der Pointer *nicht* um 2 erhöht wie in der späteren Version! Die Speicherstellen 1C8B und 1C8C sollen das folgende Hauptprogramm symbolisieren.
Bild 4 zeigt kurze Beispiele zu den immer 8 Bytes langen Pseudobefehlen für relative Sprünge und Subroutine-Aufrufe. Der Disassembler interpretiert auch die Versatzbytes als Befehle, was bitte ignoriert werden möge. Pfeile zeigen die ausgeführten Sprünge. – Im ersten Programm wird lediglich der NOP-Befehl mit einem relativen Sprung übergangen. (Das wäre auch mit einem normalen Branch-Befehl möglich, hier geht es nur ums Prinzip.) Der Versatz ist hier nur 6 Bytes groß, gezählt wird vom Byte aus, das die 06 enthält. – Im zweiten wird die Subroutine in Speicherstelle 0399, die lediglich aus einem RTS-Befehl besteht, relativ angesprungen. (Das ist mit den normalen Befehlen schon nicht mehr möglich!) – Das dritte Beispielprogramm, das in Speicherstelle 039D beginnt (nicht vorher!), zeigt einen relativen Sprung rückwärts nach 039C. Man beachte, daß bei der Addition dieses großen Versatzwertes zur Programmadresse eigentlich ein Wert von über 64 K herauskommt, der Übertrag wird jedoch ignoriert.
Im letzten Beispielprogramm mit Startpunkt in 03A7 wird die Subroutine in 03A6 rückwärts relativ aufgerufen, das Programm endet in 03AF.

```
7168  1C00  A9 68     LDA IMMED.
7170  1C02  8D B5 03  STA ABSOLUT
7173  1C05  8D B8 03  STA ABSOLUT
7176  1C08  A9 85     LDA IMMED.
7178  1C0A  8D B6 03  STA ABSOLUT
7181  1C0D  8D B9 03  STA ABSOLUT
7184  1C10  A2 01     LDX IMMED.
7186  1C12  8E B7 03  STX ABSOLUT
7189  1C15  8E BA 03  STX ABSOLUT
7192  1C18  E8        INX IMPLIED
7193  1C19  8E BA 03  STX ABSOLUT
7196  1C1C  A9 48     LDA IMMED.
7198  1C1E  8D BB 03  STA ABSOLUT
7201  1C21  8D BE 03  STA ABSOLUT
7204  1C24  A9 A5     LDA IMMED.
7206  1C26  8D BC 03  STA ABSOLUT
7209  1C29  A9 60     LDA IMMED.
7211  1C2B  8D BF 03  STA ABSOLUT
7214  1C2E  20 B5 03  JSR ABSOLUT
7217  1C31  A0 10     LDY IMMED.
7219  1C33  B1 01     LDA (IND),Y
7221  1C35  99 A5 03  STA ABS,Y
7224  1C38  C8        INY IMPLIED
7225  1C39  C0 58     CPY IMMED.
7227  1C3B  D0 F6     BNE REL 1C33
7229  1C3D  18        CLC IMPLIED
7230  1C3E  90 4B     BCC REL 1C8B

7232  1C40  08 85 00 68 85 0C 18 68
7240  1C48  69 02 85 01 68 69 00 85
7248  1C50  02 48 A5 01 48 AD 03 02
7256  1C58  C9 04 D0 03 4C 8B C3 A5
7264  1C60  0C 48 A5 00 28 60 08 48
7272  1C68  98 48 A5 01 D0 02 C6 02
7280  1C70  C6 01 A0 00 18 B1 01 65
7288  1C78  01 48 C8 B1 01 65 02 85
7296  1C80  02 68 85 01 68 A8 68 28
7304  1C88  6C 01 00 EA 60 00 00 00

7307  1C8B  EA        NOP IMPLIED
7308  1C8C  60        RTS IMPLIED
```

Bild 2. Ladeprogramm für die beiden Hilfsroutinen in Bild 1. Es ist selbst wiederum frei verschiebbar (relokatibel)

```
949   03B5  68        PLA IMPLIED
950   03B6  85 01     STA ZER PAG
952   03B8  68        PLA IMPLIED
953   03B9  85 02     STA ZER PAG
955   03BB  48        PHA IMPLIED
956   03BC  A5 01     LDA ZER PAG
958   03BE  48        PHA IMPLIED
959   03BF  60        RTS IMPLIED
```

Bild 3. Einfachstversion für das Unterprogramm AKTADR, wie sie in Bild 2 verwendet wird

```
900   0384  20 B5 03  JSR ABSOLUT
903   0387  06 00     ASL ZER PAG
905   0389  4C DB 03  JMP ABSOLUT
908   038C  EA        NOP IMPLIED
909   038D  60        RTS IMPLIED

910   038E  20 B5 03  JSR ABSOLUT
913   0391  08        PHP IMPLIED
914   0392  00        BRK IMPLIED
915   0393  20 DB 03  JSR ABSOLUT
918   0396  60        RTS IMPLIED
919   0397  EA        NOP IMPLIED
920   0398  EA        NOP IMPLIED
921   0399  60        RTS IMPLIED

924   039C  60        RTS IMPLIED
925   039D  20 B5 03  JSR ABSOLUT
928   03A0  FC        --- IMPLIED
929   03A1  FF        --- IMPLIED
930   03A2  4C DB 03  JMP ABSOLUT

934   03A6  60        RTS IMPLIED
935   03A7  20 B5 03  JSR ABSOLUT
938   03AA  FC        --- IMPLIED
939   03AB  FF        --- IMPLIED
940   03AC  20 DB 03  JSR ABSOLUT
943   03AF  60        RTS IMPLIED
```

Bild 4. Vier kurze Programmbeispiele mit den jeweils acht Bytes langen Pseudobefehlen für relative Sprünge und Unterprogramm-Aufrufe

Bild 5 zeigt einen Trace-Lauf dieses Programms, bei dem alle Befehle echt im Einzelschrittbetrieb ausgeführt und gleichzeitig disassembliert gelistet werden. An strategisch wichtigen Stellen habe ich per Tastendruck zusätzlich die Registerinhalte (im Trace-Betrieb eigener Stack mit eigenem Stack-Pointer!) abgerufen. Der in Bild 4 durch Pfeile angedeutete Ablauf wird tatsächlich so ausgeführt.

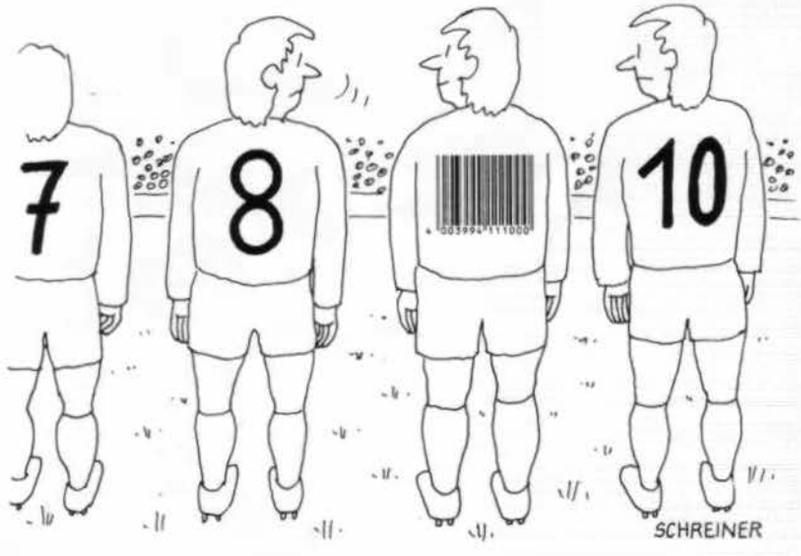
Adaption auf andere Systeme

Noch einmal: Sollen diese Routinen auf andere 6502-Rechner übertragen werden, können sie dort an eine *beliebige* Speicherstelle, sogar ins ROM (vielleicht eine Anregung für Firmen wie Commodore?) gelegt werden. Die vier in der Zero Page benötigten Speicherstellen können dort auch frei gewählt werden, solange die beiden Pointer-Bytes direkt nebeneinander liegen. Die anderen beiden Stellen müssen nicht einmal unbedingt in Zero Page liegen! Die *hier* gewählten vier Stellen in der Zero Page müßten für PETs *und* CBMs gleichermaßen verträglich sein [4].

Mit diesen Hilfsroutinen lassen sich also leicht beliebige Maschinenprogramme miteinander verbinden. Ich könnte mir sogar ein Betriebssystem ähnlich CP/M vorstellen, das Maschinenprogramme in irgendeinen freien RAM-Bereich lädt, anstartet und verwaltet. Diese Maschinenprogramme müßten nur ein paar Konventionen einhalten: Sie müßten frei verschiebbar sein (die Adressen der Hilfsroutinen im Betriebssystem lägen fest), müßten die standardisierten Ein/Ausgabe-Schnittstellen benutzen und müßten sich irgendwie über die Belegung der Zero Page einigen. Nur Letzteres erscheint schwierig, alles andere eigentlich simpel!

```
 935    03A7   20 B5 03    JSR ABSOLUT
 949    03B5   08          PHP IMPLIED
 950    03B6   85 00       STA ZER PAG
 952    03B8   68          PLA IMPLIED
 953    03B9   85 0C       STA ZER PAG
 955    03BB   18          CLC IMPLIED
 956    03BC   68          PLA IMPLIED
  A9 00 00 338 100000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
 957    03BD   69 02       ADC IMMED.
  AB 00 00 338 100000 0000 02
  AC  X  Y  SP NZCIDV  EA  EI
 959    03BF   85 01       STA ZER PAG
  AB 00 00 338 100000 0001 30
  AC  X  Y  SP NZCIDV  EA  EI
 961    03C1   68          PLA IMPLIED
  03 00 00 339 000000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
 962    03C2   69 00       ADC IMMED.
  03 00 00 339 000000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
 964    03C4   85 02       STA ZER PAG
 966    03C6   48          PHA IMPLIED
 967    03C7   A5 01       LDA ZER PAG
 969    03C9   48          PHA IMPLIED
 970    03CA   AD 03 02    LDA ABSOLUT
 973    03CD   C9 04       CMP IMMED.
 975    03CF   D0 03       BNE REL 03D4
 980    03D4   A5 0C       LDA ZER PAG
 982    03D6   48          PHA IMPLIED
 983    03D7   A5 00       LDA ZER PAG
 985    03D9   28          PLP IMPLIED
 986    03DA   60          RTS IMPLIED
  00 00 00 339 000000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
 940    03AC   20 DB 03    JSR ABSOLUT
 987    03DB   08          PHP IMPLIED
 988    03DC   48          PHA IMPLIED
 989    03DD   98          TYA IMPLIED
 990    03DE   48          PHA IMPLIED
 991    03DF   A5 01       LDA ZER PAG
 993    03E1   D0 02       BNE REL 03E5
 997    03E5   C6 01       DEC ZER PAG
 999    03E7   A0 00       LDY IMMED.
1001    03E9   18          CLC IMPLIED
1002    03EA   B1 01       LDA (IND),Y
  FC 00 00 334 100000 03AA FC
  AC  X  Y  SP NZCIDV  EA  EI
1004    03EC   65 01       ADC ZER PAG
  A6 00 00 334 101000 0001 AA
  AC  X  Y  SP NZCIDV  EA  EI
1006    03EE   48          PHA IMPLIED
  A6 00 00 333 101000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
1007    03EF   C8          INY IMPLIED
  A6 00 01 333 001000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
1008    03F0   B1 01       LDA (IND),Y
  FF 00 01 333 101000 03AB FF
  AC  X  Y  SP NZCIDV  EA  EI
1010    03F2   65 02       ADC ZER PAG
  03 00 01 333 001000 0002 03
  AC  X  Y  SP NZCIDV  EA  EI
1012    03F4   85 02       STA ZER PAG
  03 00 01 333 001000 0002 03
  AC  X  Y  SP NZCIDV  EA  EI
1014    03F6   68          PLA IMPLIED
1015    03F7   85 01       STA ZER PAG
1017    03F9   68          PLA IMPLIED
1018    03FA   A8          TAY IMPLIED
1019    03FB   68          PLA IMPLIED
1020    03FC   28          PLP IMPLIED
1021    03FD   6C 01 00    JMP ABS IND
  00 00 00 337 000000 03A6 00
  AC  X  Y  SP NZCIDV  EA  EI
 934    03A6   60          RTS IMPLIED
  00 00 00 339 000000 0000 00
  AC  X  Y  SP NZCIDV  EA  EI
 943    03AF   60          RTS IMPLIED
STACK-OVERFLOW 827
```

Bild 5. Trace-Lauf des letzten Beispielprogramms aus Bild 4. Die Meldung „Stack Overflow" am Ende ist eine normale Meldung des verwendeten Monitorprogramms und nicht durch einen Fehler bedingt



Literatur

[1] Franke, E. H.: PRTSTR – Einfache Textausgabe in Assemblerprogrammen. FUNKSCHAU-Sonderheft „Hobbycomputer 2", Franzis-Verlag.

[2] Joepgen, H. G.: 6502 simuliert neue Adressierungsart. FUNKSCHAU-Sonderheft „Programme für Kleincomputer und Taschenrechner", Franzis-Verlag.

[3] MOS Technology Inc.: 6502 Programming Manual (auch von Rockwell und Commodore erhältlich)

[4] Martin, R. und Smode, D.: ROM und RAM in PET und CBM. FUNKSCHAU-Sonderheft „Mikrocomputer-Anwendungen", Franzis-Verlag.

Reinhardt Göth

ROM-Disassembler für PET und CBM

Eines der ungelöstesten Probleme der Computerei ist das der Programmdokumentation. Manches Programm ist interessant, aber unverständlich. Manches Maschinenlisting entmutigend unkommentiert. Wer glaubt an die Wirksamkeit von nicht selbst programmierten Hexcodes? mc bringt Ihnen hier ein Programm, das wenigstens die Assemblercodes aus Hexlistings herstellt, damit Sie einem Fremdprogramm auf die Schliche kommen können.

Im Sonderheft „Mikrocomputer-Anwendungen" des Franzis-Verlages befinden sich umfangreiche ROM- und RAM-Listings u. a. für Computer der Serien PET und CBM. Nun ist es zuweilen ganz nützlich, sich zusätzlich zu den dort zu findenden Kommentaren den Speicher disassembliert aufzulisten, um genau zu sehen, was dort eigentlich passiert. Dies ermöglicht das in *Bild 1* aufgelistete Programm.

Sinnvollerweise geht man bei der Inbetriebnahme so vor, daß man das Programm sicherheitshalber vor dem Start mit RUN auf Kassette aufzeichnet.

Wird nur ein Disassembler benötigt, so können die Zeilen 170—190, 380, 390, 410—450 und 730—1010 entfallen. Ferner muß dann der Befehl USR durch PEEK ersetzt werden; Programme aus den ROMs können nun nicht mehr aufgelistet werden.

Nach dem Programmstart kann man mit den Tasten V und P wählen, ob die Ausgabe auf den Bildschirm oder auf den Drucker stattfinden soll. Kurz darauf kann man mit A, B, C, D oder E den Ausgabemodus wählen (Adressen, Byte, ASCII-Zeichen, Disassembler, Floatingpoint).

In *Bild 2* sind die für Computer mit CBM-3001-Betriebssystem nötigen Änderungen angegeben; sie werden wegen der veränderten Adressenbelegung erforderlich. *Bild 3* zeigt schließlich unterschiedliche Ausgabe-Modi.

```
170 DATA162,235,138,24,125,255,3,73,51,202,208,248,73,96,240,3,76,56,253,32,200
180 DATA214,160,,76,243,214,169,,160,,32,116,218,76,186,201
```

Bild 2. Diese Änderungen sind erforderlich, wenn man das Programm in Bild 1 auf Computern der CBM-3001-Serie laufen lassen will

```
0 REM **********************************
1 REM **** (C)  COPYRIGHT 1980 BY ****
2 REM ******** REINHARD GOETH ********
3 REM ****** BAHNHOFSTRASSE  22 ******
4 REM ****** 6465 BIEBERGEMUEND ******
5 REM **********************************


100 PRINT"V=VIDEO,P=PRINTER"
110 GETA$:IFA$="V"THENDO=3:GOTO140
120 IFA$<>"P"THEN110
130 DO=4
140 OPEN1,DO
150 DEFFNA(X)=X-(X>=10)*7+48
160 DEFFNF(Y)=Y-INT(Y)
170 DATA162,235,138,24,125,255,3,73,51,202,208,248,73,96,240,3,76,56,253,32,200
180 DATA214,160,,76,243,214,169,,160,,32,116,218,76,186,201
190 FORI=826TO862:READN:POKEI,N:NEXT:POKE1,58:POKE2,3
200 DATABRK,1,ORA,9,,,,ORA,3,ASL,3,PHP,1,ORA,13,ASL,2,,,ORA,6,ASL,6
210 DATABPL,12,ORA,10,,,,,ORA,4,ASL,4,CLC,1,ORA,8,,,,,ORA,7,ASL,7
220 DATAJSR,6,AND,9,,,BIT,3,AND,3,ROL,3,PLP,1,AND,13,ROL,2,BIT,6,AND,6,ROL,6
230 DATABMI,12,AND,10,,,,,AND,4,ROL,4,SEC,1,AND,8,,,,,AND,7,ROL,7
240 DATARTI,1,EOR,9,,,,,EOR,3,LSR,3,PHA,1,EOR,13,LSR,2,JMP,6,EOR,6,LSR,6
250 DATABVC,12,EOR,10,,,,,EOR,4,LSR,4,CLI,1,EOR,8,,,,,EOR,7,LSR,7
260 DATARTS,1,ADC,9,,,,,ADC,3,ROR,3,PLA,1,ADC,13,ROR,2,JMP,11,ADC,6,ROR,6
270 DATABVS,12,ADC,10,,,,,ADC,4,ROR,4,SEI,1,ADC,8,,,,,ADC,7,ROR,7
280 DATA,,STA,9,,,STY,3,STA,3,STX,3,DEY,1,,,TXA,1,STY,6,STA,6,STX,6
290 DATABCC,12,STA,10,,,STY,4,STA,4,STX,5,TYA,1,STA,8,TXS,1,,,STA,7,,
300 DATALDY,13,LDA,9,LDX,13,LDY,3,LDA,3,LDX,3,TAY,1,LDA,13,TAX,1,LDY,6,LDA,6,LDX,6

310 DATABCS,12,LDA,10,,,LDY,4,LDA,4,LDX,5,CLV,1,LDA,8,TSX,1,LDY,7,LDA,7,LDX,8
320 DATACPY,13,CMP,9,,,CPY,3,CMP,3,DEC,3,INY,1,CMP,13,DEX,1,CPY,6,CMP,6,DEC,6
330 DATABNE,12,CMP,10,,,,CMP,4,DEC,4,CLD,1,CMP,8,,,,CMP,7,DEC,7
340 DATACPX,13,SBC,9,,,CPX,3,SBC,3,INC,3,INX,1,SBC,13,NOP,1,CPX,6,SBC,6,INC,6
350 DATABEQ,12,SBC,10,,,,SBC,4,INC,4,SED,1,SBC,8,,,,SBC,7,INC,7
360 DIMA$(191),A%(191)
370 FORI=0TO191:READA$(I),A%(I):NEXT
380 PRINT#1,:PRINT"A=ADD,B=BYT,C=CHR,D=DIS,E=MMB"
390 GETZ$:OO=ASC(Z$+" ")-65:IFOO<0OR4<OOTHEN390
400 INPUT"GEBEN SIE ANFANGS- UND ENDADRESSE EIN ";A,E:CMD1,;
410 ONOOGOTO730,700,460,980
420 IFA>ETHEN1050
430 GOSUB1020:P=USR(A+1):GOSUB710
440 PRINTSPC(7)"$";:GOSUB710:P=USR(A):GOSUB710:PRINT" =";
450 P=USR(A)+256*USR(A+1):PRINTSPC(7-LEN(STR$(P)));P:A=A+2:GOTO420
460 IFA>ETHEN1050
470 GOSUB1020
480 K=USR(A)/4:R=A+1:IFK-INT(K)=.75THENPRINTSPC(9)"???":GOTO460
490 K=K*3+.5:IFA$(INT(K))="*"THENPRINTSPC(9)"???":GOTO460
500 ONA%(K)GOSUB680,680,690,690,690,700,700,700,690,690,700,690,690
510 PRINTA$(K)" ";:ONA%(K)GOSUB530,540,550,560,570,580,590,600,610,620,630,640,670
520 GOTO460
530 PRINT:RETURN
540 PRINT"A":RETURN
550 P=USR(A):GOSUB710:A=A+1:PRINT:RETURN
560 P=USR(A):GOSUB710:PRINT",X":A=A+1:RETURN
570 P=USR(A):GOSUB710:PRINT",Y":A=A+1:RETURN
580 Q=USR(A)+256*USR(A+1):GOSUB720:PRINT:A=A+2:RETURN
590 Q=USR(A)+256*USR(A+1):GOSUB720:PRINT",X":A=A+2:RETURN
```

Bild 1. Listing des in Basic geschriebenen Programms zum Untersuchen von ROM- und RAM-Bereichen für den PET-2001. Wichtig: Auch die REM-Zeilen dürfen nicht fehlen!

a

```
C000 49152 1D C7      #C71D =  50973
C002 49154 48 C6      #C648 =  50760
C004 49156 35 CC      #CC35 =  52277
C006 49158 EF C7      #C7EF =  51183
C008 49160 C5 CA      #CAC5 =  51909
```

b

```
DDE8 56800 FA         #FA   =  250
DDE9 56809 0A         #0A   =   10
DDEA 56810 1F         #1F   =   31
DDEB 56811 00         #00   =    0
DDEC 56812 00         #00   =    0
DDED 56813 98         #98   =  152
DDEE 56814 96         #96   =  150
DDEF 56815 80         #80   =  128
```

c

```
E18E 57742 42         B
E18F 57743 41         A
E190 57744 53         S
E191 57745 49         I
E192 57746 43         C
E193 57747 20         SPACE
E194 57748 2A         *
E195 57749 2A         *
E196 57750 2A         *
E197 57751 11         ■CURSOR DOWN■
E198 57752 11         ■CURSOR DOWN■
E199 57753 11         ■CURSOR DOWN■
E19A 57754 00         ■END OF BLOCK■
```

d

```
F53B 62779 D0 3E      BNE F57B
F53D 62781 AE 62 02   LDX 0262
F540 62784 A0 00      LDY #00
F542 62786 8C 0C 02   STY 020C
F545 62789 E0 0A      CPX #0A
F547 62791 F0 F2      BEQ F53B
```

e

```
E01A 57370 81 490FDAA2   1.57079633
E01F 57375 83 490FDAA2   6.28318531
E024 57380 7F 00000000    .25
E029 57385 05 84E61A2D -4.89193226E-38
E02E 57390 1B 86280AFB -2.06700619E-31
E033 57395 F8 87996889 -7.04071567E+35
```

Bild 3. Ausgabeformate des Programms: A = Adressenmodus, B = Bytemodus, C = Zeichenmodus, D = Disassembler, E = Gleitkommazahlen-Modus

```
600 D=USR(A)+256*USR(A+1):GOSUB720:PRINT",Y" A=A+2:RETURN
610 PRINT"(";P=USR(A):GOSUB710:PRINT",X)" A=A+1:RETURN
620 PRINT"(";P=USR(A):GOSUB710:PRINT"),Y" A=A+1:RETURN
630 PRINT"(";D=USR(A)+256*USR(A+1):GOSUB720:PRINT")" A=A+2:RETURN
640 A=A+1:IFUSR(A-1)<128THEND=A+USR(A-1):GOSUB720:PRINT:RETURN
650 D=USR(A-1)+A-256:IFD<0THEND=D+65536
660 GOSUB720:PRINT:RETURN
670 PRINT"#";P=USR(A):A=A+1:GOSUB710:PRINT:RETURN
680 PRINTSPC(9);:RETURN
690 P=USR(A):GOSUB710:PRINTSPC(7);:RETURN
700 P=USR(A):GOSUB710:PRINT" ";:P=USR(A+1):GOSUB710:PRINTSPC(4);:RETURN
710 PRINTCHR$(FNA(P/16))CHR$(FNA(FNF(P/16)*16));:RETURN
720 P=D/256:GOSUB710:P=256*FNF(P):GOTO710
730 IFA>ETHEN1050
740 GOSUB1020
750 PRINTSPC(9)"#";:GOSUB710
760 PRINT"  ="SPC(5-LEN(STR$(P)))P
770 A=A+1:GOTO730
780 IFA>ETHEN1050
790 GOSUB1020
800 PRINTSPC(9);
810 IFUSR(A)=13THENPRINT"■ ■CAR. RETURN■ ";:GOTO970
820 IFUSR(A)=10THENPRINT"■ ■LINE FEED■ ";:GOTO970
830 IFUSR(A)=17THENPRINT"■ ■CURSOR DOWN■ ";:GOTO970
840 IFUSR(A)=145THENPRINT"■ ■CURSOR UP■ ";:GOTO970
850 IFUSR(A)=29THENPRINT"■ ■CURSOR RIGHT■ ";:GOTO970
860 IFUSR(A)=157THENPRINT"■ ■CURSOR LEFT■ ";:GOTO970
870 IFUSR(A)=19THENPRINT"■ ■CURSOR HOME■ ";:GOTO970
880 IFUSR(A)=147THENPRINT"■ ■CLEAR SCREEN■ ";:GOTO970
890 IFUSR(A)=18THENPRINT"■ ■REVERS ON■ ";:GOTO970
900 IFUSR(A)=146THENPRINT"■ ■REVERS OFF■ ";:GOTO970
910 IFUSR(A)=0THENPRINT"■ ■END OF BLOCK■ ";:GOTO970
920 IFUSR(A)>127THENPRINT"■ ■";
930 IF(127ANDUSR(A))=32THENPRINT"SPACE":GOTO970
940 IF(127ANDUSR(A))=96THENPRINT"? SPACE":GOTO970
950 IF(127ANDUSR(A))<32THENPRINT"???":GOTO970
960 PRINTCHR$(127ANDUSR(A))
970 A=A+1:GOTO780
980 IFA>ETHEN1050
990 POKE856,A/256:POKE854,A-INT(A/256)*256
1000 GOSUB1020:FORA=A+1TOA+3:P=USR(A):GOSUB710:NEXT:PRINT" ";
1010 SYS853:GOTO980
1020 D=A:GOSUB720:P=USR(A)
1030 PRINTSPC(7-LEN(STR$(A)))MID$(STR$(A),2)" ";
1040 GOSUB710:PRINT" ";:RETURN
1050 PRINT#1,:PRINT"WEITER (J/N)"
1060 GETA$:IFA$="J"THEN380
1070 IFA$<>"N"THEN1060
1080 CLOSE1
1090 END
```

Sharp MZ-80B, eine Neuheit

Auf der Hannover-Messe präsentierte die Firma Sharp eine aufregende Neuheit: den Sharp MZ-80B. Das Aufregende an ihm ist, daß er nur einen Urlader von 2 KByte Länge als ROM besitzt. Im Gehäuse können bis zu 64 KByte herrlich freie dynamische RAM installiert werden. Alle Sprachen werden über den Urlader eingelesen. Das ergibt maximale Gestaltungsfreiheit für den Benutzer.
Die vorläufigen technischen Daten:

Zentraleinh. (CPU):	Z-80A
Speicher:	ROM: 2 KByte RAM: 32 KByte dynamisch (im Gerät bis zu 64 KByte ausbaufähig)
Bildschirm:	grüne 25-cm-Bildröhre, 8 × 8-Punktmatrix, 40/80 Zeichen × 25 Zeilen (umschaltbar)
Kassette:	mit Software-Steuerung oder von Hand, Normalkassette, Datenübertragungsgeschwindigkeit 2400 Bit/s, Übertragungssystem Sharp-PWM
Tastatur:	ASCII-Tastatur, Schriftzeichen (Groß- und Kleinbuchstaben), Grafiksymbole, 10 Sondertasten, Zehnertastatur
Editing:	Cursorsteuerung (oben, unten, links, rechts, Ausgangsstellung), Einfügen, Löschen
Zeitfunktion (Uhr):	integriert
Musikfunktion:	integriert
Stromversorgung:	220 V, 50/60 Hz, Leistungsaufnahme 50 W (ohne Zusatzgeräte)
Temperatur:	Betrieb: 0...35 °C Lagerung: −15...+60 °C
Luftfeuchtigkeit:	Unter 80 % (im Betrieb)
Gewicht:	ca. 16 kg
Abmessungen:	450 (B) × 520 (T) × 270 (H) mm³.

Ing. (grad.) Bernd Sundermeier

Lohn- und Einkommensteuer mit dem HP-67

Im Franzis-Sonderheft Nr. 31, „Programme für Kleincomputer und Taschenrechner", ist ein Programm für den UPN-Rechner HP-67 vorgestellt worden, mit dem die Lohn- und Einkommensteuern für die Jahre 1979 und 1980 berechnet werden können. Aufgrund des vom Bundestag im August 1980 beschlossenen Steuerentlastungsgesetzes sind einige Änderungen des Programms erforderlich, um es auch für 1981 und 1982 einsetzen zu können.

Für die Berechnungen der Einkommensteuer ab 1981 sind die folgenden Gleichungen und Geltungsbereiche anzuwenden:
Für zu versteuernde Einkommen bis 4212 DM:

$$EST = 0 \quad (1)$$

Für zu versteuernde Einkommen von 4213 DM bis 18 000 DM:

$$EST = 0{,}22x - 926 \quad (2)$$

Für zu versteuernde Einkommen von 18 001 DM bis 59 999 DM:

$$EST = (((3{,}05y - 73{,}76)\,y + 695)\,y + 2200)\,y + 3034 \quad (3)$$

Für zu versteuernde Einkommen von 60 000 DM bis 129 999 DM:

$$EST = (((0{,}09z - 5{,}45)\,z + 88{,}13)\,z + 5040)\,z + 20018 \quad (4)$$

Für zu versteuernde Einkommen von mehr als 130 000 DM:

$$EST = 0{,}56x - 14837 \quad (5)$$

„x" ist das abgerundete zu versteuernde Einkommen.

$$y = (x - 18000)/10000 \quad (6)$$

$$z = (x - 60000)/10000 \quad (7)$$

Die Rundung des zu versteuernden Einkommens wird nicht mehr auf den nächsten durch 30 bzw. 60 teilbaren Betrag, sondern generell auf den nächstniedrigeren ohne Rest durch 54 teilbaren Betrag durchgeführt. Hierdurch ergibt sich auch bei der Berechnung der Lohnsteuer ein anderer Rundungsmodus für das zu versteuernde Monatseinkommen. Die neuen Monatslohnsteuertabellen haben eine Sprungweite von 4,50 DM, so daß die Steuersätze jeweils für einen Bereich von 4,50 DM (eine Tabellenstufe) konstant sind. Das zu versteuernde Monatseinkommen „M" wird laut folgender Gleichung abgerundet und auf das entsprechende Jahreseinkommen „A" umgerechnet:

$$A = (INT[(M - 1{,}5)/4{,}5) \times 4{,}5 + 1{,}5) \times 12 \quad (8)$$

Die Vorsorgepauschale „VP" wird berechnet zu:

$$VP = 0{,}09 \times (A + 53) \quad (9)$$

Schließlich und endlich sind auch die bei der Lohnsteuerberechnung zu berücksichtigenden Freibeträge geändert worden. Die Summen der neuen Freibeträge für die verschiedenen Steuerklassen sind in *Tabelle 1* angegeben. Der Freibetrag in Steuerklasse 2 bei Berücksichtigung von einem oder mehreren Kindern ($K \neq 0$) wird 1982 um 1212 DM nochmals angehoben. Außerdem werden dann auch die Höchstbeträge für die Vorsorgepauschale von 2100 DM auf 2340 DM, bzw. von 4200 DM auf 4680 DM und von 1050 DM auf 1170 DM erhöht.

Durch die Gesetzesänderungen hat die Berechnung der Lohnsteuer zwar kaum an Umfang verloren, es konnten jedoch u. a. durch das etwas vereinfachte Rundungsverfahren einige Programmschritte eingespart werden, so daß mit dem neuen Programm neben der Berechnung der Einkommensteuer nach Grund- und Splittingtarif auch die Berechnung der Lohnsteuer für alle Steuerklassen ohne Umweg (wie bisher bei Steuerklasse 2) möglich ist. Das „alte" Programm ist in *Bild 1*, die geänderten Programmschritte in *Bild 2*, die neue Belegung der Speicherregister ist im *Bild 3* dargestellt. Angegeben sind im Bild 2 jeweils in der linken Spalte die zu löschenden Schritte des ursprünglichen Programms und in der rechten Spalte die neuen Programmschritte, deren endgültige Schrittnummern (nach Ausführung aller Änderungen) in der mittleren Spalte vermerkt sind. Zum Ändern des Programms löscht man am besten erst alle in der linken Spalte angegebenen alten Programmschritte, von hinten beginnend. Dann werden, von vorn beginnend, die neuen Schritte eingefügt. Wenn die ersten 36 neuen Schritte eingetippt sind, springt man mit „GTO.079" auf Schritt 79 und kann dann die Schritte 80 und 81 einfügen usw.

Um Speicherplatz einzusparen, ist bei der neuen Version die Eingabe für die Lohnsteuerberechnung geändert worden. Die Eingabe in das Stack-Register

Tabelle 1: Summen der Freibeträge ab 1981

Steuerklasse	Σ der FB
1	1314
2 (K = 0)	2178
2 (K ≠ 0)	4338*
3	1584
4	1314
5	1044
6	18

*ab 1. 1. 1982: 5550

```
001  LBL A      034  LBL c      067  LBL 6      100  GOTO 8     133    2      167  STO B      196    1
002    .        035  ST I       068  RCL A      101  GSB d      134    :      168  RCL A      197    2
003    5        036  RCL 8      069    2        102  X=Y        135  RTN      169    4        198    -
004    +        037  LBL 4'     070    4        103  GOTO 9     136  LBL B      170    .      199  GOTO 5'
005    .        038  STO x 9    071  LBL 3'     104  R↓         137  F? 0       171    8      200  LBL 7'
006    4        039  RCL 7      072    -        105    2        138  GOTO B'    172    -      201  RCL 9
007    x        040  STO + 9    073  SF 1       106    x        139  RCL 2      173  X>0      202  GOTO e
008  INT        041  F? 2       074  GSB B      107  LBL E      140    1        174  STO B      203  LBL 8'
009    .        042  RTN        075  CF 1       108  SF 1       141    0        175    4        204    .
010    4        043  GSB 7      076  SF 0       109  GSB B      142    x        176  ST I       205    1
011    :        044  GSB B      077  GSB d      110  CF 1       143  X≠Y        177  LBL a      206  P≠S
012  RCL 5      045  LBL D      078  STO 9      111  RC I       144  X=Y        178  ISZ        207  SF 2
013    +        046    1        079    3        112    +        145  SF 2       179  RCL i      208  LBL e
014    1        047    .        080    x        113  RCL A      146    6        180  RCL A      209    4
015    2        048    2        081  GSB C      114  X≠Y        147    0        181  X=Y        210  ST I
016    x        049    :        082  RCL 9      115    -        148  F? 2       182  GOTO i     211  X≠Y
017  STO A      050  INT        083    5        116  RTN        149  GSB d      183  RCL 8      212  LBL b
018    9        051    1        084    x        117  LBL 8      150    :        184  X>Y        213  RCL B
019    %        052    0        085  X≠Y        118    1        151  LST X      185  GOTO a     214    x
020  STO B      053    :        086  STO 9      119    .        152  X≠Y        186  X≠Y        215  RCL i
021  R↑         054  R/S        087  X≠Y        120    5        153  INT        187  RCL 0      216    +
022  ST I       055  LBL 3      088  GSB C      121    x        154    x        188    x        217  DSZ
023  R↑         056  SF 2       089  RCL 9      122  GOTO E     155  F? 1       189  RCL E      218  GOTO b
024  STO 9      057  RCL 6      090    -        123  LBL 9      156  RTN        190    -        219  F?  2
025  GOTO i     058  GSB c      091  CF 0       124    +        157  LBL B'     191  GOTO 5     220  P≠S
026  LBL 4      059  STO + 9    092  GOTO D     125  GOTO E     158  P≠S        192  LBL 6'     221  LBL 5'
027  RCL 0      060  GSB 7      093  LBL 7      126  LBL C      159  EEX        193  RCL 2      222  P≠S
028  ST I       061  GSB C      094  RCL B      127  GSB d      160    4        194    x        223  INT
029  RCL 8      062  GOTO D     095  RCL C      128  GSB B      161    :        195    8        224  RTN
030  GSB d      063  LBL 5      096  X>Y        129    2        162  STO A
031  GOTO 4'    064  RCL A      097  X≠Y        130    x        163    1
032  LBL 1      065  RCL D      098  RCL 9      131  RTN        164    .
033  RCL 0      066  GOTO 3'    099  X=Y        132  LBL d      165    6
                                                               166    -
```

Bild 1. Listing des ursprünglichen HP-67-Programms. Es ist so leider nicht mehr verwendbar, da sich die gesetzlichen Berechnungsgrundlagen verändert haben

```
001  001  LBL A      019  019  STO A      069  080    1      149  ---  ---
002  002  STO 9      020  020    5        070  081    8
003  003  R↓         021  021    3        118  129    3      163  164  RCL 6
004  004  ST I       022  022    +        119  130  GSB d    164  ---  ---
005  005  R↓         023  023    9        120  ---  ---      165  ---  ---
006  006    3        024  024    %
007  007  GSB d      025  025  STO B      139  149    5      169  168    6
008  008    -        ---  026  GTO i      140  150    4      170  ---  ---
009  009    9        ---  027  LBL 2      141  ---  ---      171  ---  ---
010  010  GSB d      ---  028  RCL 9      142  ---  ---
011  011    :        ---  029  X>0        143  ---  ---      195  192    9
012  012  INT        ---  030  RCL 5      144  ---  ---      196  193    2
013  013    5        ---  031    2        145  ---  ---      197  194    6
014  014    4        ---  032    1        146  ---  ---
015  015    x        ---  033    7        147  ---  ---      205  202    0
016  016    1        ---  034    8        148  ---  ---      ---  203    9
017  017    8        ---  035    +
018  018    +        ---  036  GTO c
```

Bild 2. Notwendige Änderungen des Programms in Bild 1; links steht die ursprüngliche Zeilennummer, in der Mitte die neue, und rechts der neue Befehl

erfolgt jetzt in der Reihenfolge Monatseinkommen, Steuerklasse, Anzahl der Kinder laut Steuerkarte. Das neue Programm kommt trotz des Einfügens der Berechnung von Steuerklasse 2 mit nur 222 Programmschritten aus. Der Vollständigkeit halber sei hier jedoch noch erwähnt, daß für die Steuerklassen 5 und 6 bei geringen Einkommen ein Mindeststeuersatz erhoben wird. Dieser beträgt bei Steuerklasse 5:

MIST5 = INT(INT((A – 1044) x 0,22)/1,2)/10 (10)

und für Steuerklasse 6:

MIST6 = INT(INT((A – 18) x 0,22)/1,2)/10 (11)

Für 1981 ergibt sich eine Anwendung der Mindeststeuersätze in Steuerklasse 5 bis zu 554,99 DM und in Steuerklasse 6 bis zu 469,49 DM Monatseinkommen. Dies wird durch das Programm mangels Speicherplatz nicht berücksichtigt.

Bild 3. Auch die Belegung der Arbeitsspeicher hat sich gegenüber Bild 1 etwas verändert

I	: Arbeitsspeicher	R_C	: 4752 ***
R_A	: Arbeitsspeicher	R_D	: 1044
R_B	: Arbeitsspeicher	R_E	: 14837
R_0	: 1314	R_{S0}	: 5600
R_1	: 20018	R_{S1}	: 3034
R_2	: 5040	R_{S2}	: 2200
R_3	: 88,13	R_{S3}	: 695
R_4	: -5,45	R_{S4}	: -73,76
R_5	: 2160 *	R_{S5}	: 0,4213
R_6	: 1584	R_{S6}	: 1,8
R_7	: 2100 **	R_{S7}	: 6
R_8	: 600	R_{S8}	: 13
R_9	: Arbeitsspeicher	R_{S9}	: 3,05

* ab 1.1.82: 3372
** ab 1.1.82: 2340
*** 4752 = 0,09 x 52800 = 9% der Beitragsbemessungsgrenze 1981 für die gesetzliche Rentenversicherung. Ab 1.1.82 ist hier der Betrag für 1982 einzusetzen.

TI-59-Besonderheiten

Nach [1] holt der Mikroprozessor im TI 59 die Befehle aus dem Programmspeicher nicht einzeln, sondern in Gruppen zu je acht, also z. B. Programmspeicherstellen 0 bis 7, dann 8 bis 15 usw. Diese acht Programmschritte werden nacheinander abgearbeitet, soweit kein Sprungbefehl o. ä. die normale Befehlsabfolge ändert. Man kann sich dies beim TI 59 zunutze machen, indem man im Programm einen Lesebefehl für eine Magnetkarte verwendet, z. B. 1 INV WRT, und dann im gleichen (!) Achterblock mit einem Sprungbefehl wie GTO A oder GTO 002 fortfährt. Selbst wenn diese Befehlsfolge durch die eingelesene Magnetkarte überschrieben wird, wird noch der Sprungbefehl ausgeführt.

Karte 1 laden mit und A starten. Dann Karte 2 einlesen. Der Rechner hält mit dem Wert 3 in der Anzeige. Verschiebt man dagegen das Programm der Karte 1 um drei Schritte nach hinten, so daß der WRT-Befehl an die Stelle 007 rutscht, so führt das obige Vorgehen dazu, daß nach dem Einlesen von Karte 2 als erstes der Programmschritt 008 ausgeführt wird und der Rechner mit blinkender Anzeige hält. Diese Besonderheit wurde in [2] übersehen, wo dieses Verfahren auch beschrieben wurde.

Gelegentlich will man in einem (umfangreichen) Programm eine Befehlsfolge wie z. B.

LBL A + RCL IND 01 × RCL IND 02 OP 31 DSZ 02 A =

durchführen, ohne die Register 01 und 02 zur Verfügung zu haben. Die Verminderung (oder Vergrößerung) eines Zeigers wie im Register 01 durch OP 31 ist aber nur für die Register 00 bis 09 möglich. Will man statt der Register 01 und 02 z. B. die Register 51 und 52 verwenden, so kann man die obige Befehlsfolge ersetzen durch

LBL A + RCL IND 51 × 1 +/− SUM 51 RCL IND 52 DSZ 52 A =.

Der Trick dabei ist, daß durch das Zeichen „x" die bisherige Summe und der Faktor „RCL IND 51" in die Hilfsregister eingeschrieben werden (vgl. z. B. [3]). Die durch „1 +/− Sum 51" entstehende „−1" in der Anzeige wird dann durch „RCL IND 52" überschrieben und die Rechnung wird richtig fortgesetzt. (Die Verwendung des DSZ-Befehls in Verbindung mit allen Registern außer dem Register Nr. 40 wurde z. B. in [3] beschrieben).

Beispiel:

Karte 1, Block 1

000	76	LBL
001	11	A
002	01	1
003	22	INV
004	96	WRT
005	61	GTO
006	11	A
007	91	R/S
008	00	0
009	00	0
010	00	0

Karte 2, Block 1

000	76	LBL
001	11	A
002	03	3
003	91	R/S
004	00	0
005	00	0
006	00	0
007	00	0
008	00	0
009	35	1/X
010	91	R/S

Dr. Harald Lindner

Literatur

[1] *Göth, Jungbauer* und *Müller:* Unbekannte Hardwareeigenschaften des TI 59, Display (herausgegeben von H. Schnepf, Buchenweg 24, Köln), Nr. 5/6, S. 76 (1978)

[2] *Panzer:* TI-59-Kartentricks, Chip Nr. 3, S. 14 (1980)

[3] *Jäkel:* Wenn der Speicher überläuft. TI-59-Sparmaßnahmen. FUNKSCHAU Nr. 10, S. 97–99 (1980)

Mikrocomputer

Fachbücher zum Thema: Geräte, Begriffe, Befehle, Programme

★ **Neuerscheinung**

Der Ein-Chip-Mikrocomputer.

Fortschritt durch moderne Technologie, Anwendung und Programmierung. Von Horst **Pelka.** – 141 Seiten, 50 Abbildungen und 19 Tabellen.
Lwstr-kart. DM 16.80
ISBN 3-7723-6831-X

Der Schwerpunkt der Darstellung liegt ganz klar auf der Programmierung. Dabei geht der Autor die Variationen in den Befehlssätzen der 8048-Familie durch und schildert die Architektur der einzelnen Typen. Der nächste Schritt zeigt dann, wie groß der Programmierspielraum ist und damit ergeben sich gleich die verschiedensten Anwendungsmöglichkeiten. – Die eingestreuten Standardprogramme sind für die fortgeschrittenen Elektroniker und Informatiker das erwartete Rüstzeug im Umgang mit Ein-Chip-Mikrocomputern.

★ **Neuerscheinung**

Mikrocomputer-Hard- und Softwarepraxis.

Anhand ausführlicher Beispiele und größerer Programme wird das Programmieren immer perfekter. Von Rolf Dieter **Klein.** – 220 Seiten, 125 Abbildungen, 6 Tabellen.
Lwstr-geb. DM 38.–
ISBN 3-7723-6811-5

Die Erfahrungen, die dieser Band enthält, stammen aus der Praxis im eigenen Labor, in welchem die Hardware aufgebaut und mit der erforderlichen Software versehen wurde. Alle modernen Peripheriegeräte wurden dabei, und teilweise durch eigene BUS-Systeme, mit Programmen verknüpft. Die umfangreichen Beschreibungen von EDITOR, MONITOR, RECOLATOR, TRACER und ASSAMBLER machen das Buch besonders interessant.

ABC der Mikroprozessoren und Mikrocomputer.

Neue Fachwörter und Abkürzungen für Elektroniker, Programmierer und Praktiker verständlich gemacht. Von Horst **Pelka.** – 159 Seiten mit 45 Abbildungen. (RPB electronic-taschenbuch Nr. 135).
Kart. DM 10.80
ISBN 3-7723-1351-5

★ **Neuerscheinung**

Basic für Mikrocomputer.

Geräte – Begriffe – Befehle – Programme. Von Herwig **Feichtinger.** – 232 Seiten, 40 Abbildungen.
Lwstr-kart. DM 26.–
ISBN 3-7723-6821-2

Dieses praxisorientierte Buch ist Einführung und Nachschlagewerk zugleich: Begriffe aus der Computer-Fachsprache wie ASCII, RS-232-Schnittstelle oder IEC-Bus werden ebenso ausführlich erläutert, wie alle derzeit üblichen Befehlsworte der Programmiersprache Basic. Marktübliche Basic-Rechner werden einander gegenübergestellt – einerseits, um vor dem Kauf die Qual der Wahl zu erleichtern, andererseits, um das Anpassen von Programmen an den eigenen Rechner zu ermöglichen. Und schließlich findet der Leser handfeste Tips für das Erstellen eigener Programme und Beispiele fertiger Problemlösungen für typische Anwendungsfälle.

Von der Schaltalgebra zum Mikroprozessor.

Die Mikroprozessoren und ihre festverdrahtete und programmierbare Logik. Von Horst **Pelka.** – 2., verbesserte und erweiterte Auflage. 339 Seiten, 178 Abbildungen, 24 Tabellen.
Lwstr-kart. DM 28.–
ISBN 3-7723-6422-5

Die mathematische Logik und die elektronische Technik, die den Mikroprozessor/Mikrocomputer ausmachen, sind hier in ihren Grundlagen dargestellt. Fast die Hälfte des Bandes befaßt sich mit dem Programmieren. Hervorzuheben ist eine Vergleichstabelle der Befehlssätze von acht Mikroprozessoren.

★ **Neuauflage**

Praxis mit Mikroprozessoren.

Wie herkömmliche Digitalschaltungen durch Mikroprozessoren erweitert, ausgebaut oder ersetzt werden können. Von Horst **Pelka.** – 2., neu bearbeitete und erweiterte Auflage. 303 Seiten, 162 Abbildungen, 4 Tabellen.
Lwstr-kart. DM 28.–
ISBN 3-7723-6582-5

Mikrocomputersystem (Hardware), Peripherie, Zusatz-, Hilfs- und Prüfschaltungen, Software. Eine Prom-Programmier-Einrichtung. Einrichtung zum Überspielen von Mikroprozessor-Programmen auf Kassettenrecorder. Kompatibilität mit anderen Systemen. Hinweise zur Fädeltechnik.

Mikrocomputersysteme.

Selbstbau – Programmierung – Anwendung. Von Rolf-Dieter **Klein.** – 2., verbesserte Auflage, 160 Seiten, 133 Abbildungen und 11 Tabellen.
Lwstr-geb. DM 32.–
ISBN 3-7723-6382-2

Kaum zu glauben, daß ein Mikrocomputer im Selbstbau hergestellt werden kann! Daß dieses Vorhaben glückte, hat der Autor bewiesen. Wie ein hinreichend ausgebildeter Elektroniker das nachvollziehen kann, wird in dem Buch hier dargestellt.

Als Abschluß und Höhepunkt fügt der Autor Anregungen hinzu, selbst Programme zu schreiben und in dem eigenen Mikrocomputer zu erproben. Was will man mehr?

Anwendungsbeispiele für den Mikroprozessor 6502.

Hardware-Tips und nützliche Programmbeispiele in Maschinensprache. Von Herwig **Feichtinger.** – 96 Seiten mit 40 Abbildungen. (RPB electronic-taschenbuch Nr. 173).
Kart. DM 8.80
ISBN 3-7723-1731-6

★ **Neuerscheinung**

Z-80-Applikationsbuch.

Eine Sammlung von gut dokumentierten Programmen, die universell anwendbar sind. Von Michael **Klein.** – 144 Seiten, 89 Abbildungen.
Lwstr-geb. DM 32.–
ISBN 3-7723-6671-6

Das Applikationsbuch ist ein Schritt in die Richtung „Lösen von Standardproblemen". Dem Benutzer werden einige fertige Programme angeboten und erklärt, anhand der er entweder seine eigenen Probleme verstehen und bewältigen kann, oder die er zumindest teilweise übernehmen kann.

Bewußt legt der Autor den Schwerpunkt auf die sofortige Anwendbarkeit der dargestellten Methoden und Programme. Er bietet Standardlösungen z. B. für die Ein-/Ausgabe über eine serielle Schnittstelle oder Interruptschaltung, führt Programme zur Meßwertverarbeitung an und zeigt, wie man sich nützliche Arbeitshilfen schaffen kann.

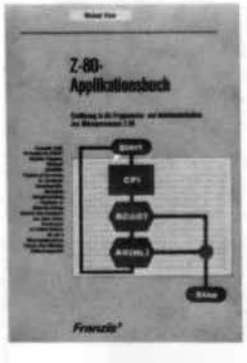
Alles in allem: Warum soll denn jeder von vorne anfangen und immer dasselbe separat „erfinden"? Das muß nicht sein, denn jetzt gibt es das Z-80-Applikationsbuch.

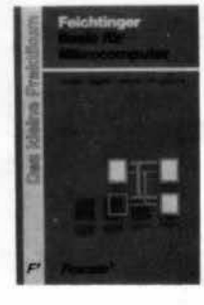
★ **Neuerscheinung**

Schlüssel zum Mikrocomputer.

Der Schritt vom Mikroprozessor zum Mikrocomputer. Von Ing. Dieter **Nührmann.** – 224 Seiten, 154 Abbildungen, zahlreiche Tabellen.
Lwstr-geb. mit Schutzumschlag
DM 34.–
ISBN 3-7723-6781-X

Das Sachbuch gibt dem unbefangenen Leser eine grundsolide Starthilfe in die nüchterne und sachliche Welt der Mikrocomputer. Imponierend ist der Stil mit dem der Elektroniker Dieter Nührmann das Innenleben dieser Apparate auseinanderfädelt, damit jedermann sie versteht und begreift.





Franzis-Bücher erhalten Sie durch jede Buchhandlung sowie in den einschlägigen Fachhandlungen. Bestellungen auch an den Verlag.

Franzis' der große Fachverlag für angewandte Elektronik

Friedhelm Schiersching

TRS-DOS mit Auto-Start

Um den Mikrocomputer TRS-80 mit seinem Disketten-Betriebssystem TRS-DOS „hochzufahren", sind eine ganze Menge Handgriffe und Tastendrücke erforderlich. Einige kleine maschinensprachliche Kniffe helfen, den Aufwand bei der Inbetriebnahme des Systems auf ein Minimum zu reduzieren.

Zunächst ist der TRS-80-Besitzer über seine Neuerwerbung, ein oder zwei Diskettenlaufwerke, glücklich: Zu Recht, denn viele der Mängel des Kassettenbetriebes existieren nun nicht mehr. Die grauen Haare, die sich langsam wegen der langen Ladezeiten von der Kassette – wenn das Laden überhaupt funktionierte – eingestellt hatten, verschwinden wieder. Keine Ladeprobleme, schnelles Laden.

Doch der Mensch ist nie zufrieden. Da bleibt doch auch bei dem Betrieb mit der Diskette immer noch eine sich ständig wiederholende Tätigkeit, die zwar nicht viel Zeit kostet, doch sie ist da und stört nach einiger Zeit.

DOS-Start von Hand

Wird die Diskette gestartet, erscheint nach einiger Zeit DOS READY. Nun wiederholt sich immer die gleiche Eingabe:
BASIC (Enter), warten, 3 (oder eine andere Zahl), (Enter), 65529 (Enter) oder auch hier eine andere Zahl. Und dann LOAD"..." oder RUN"..." und das Programm. Mit der Zeit stört diese Tipperei.

Selbstverständlich hat das TRS-DOS-System die Möglichkeit, mit AUTO einen kleinen Durchstart zu vollziehen. Nach der Eingabe AUTO BASIC läuft dann die Diskette bis zu „HOW MANY FILES?" durch, doch nicht weiter. Die nächsten Angaben müssen wieder eingetippt werden.

Andere Betriebs-Systeme haben die Möglichkeit, den Start bis zum Beginn des Programms anzugeben, z. B.: AUTO BASIC,3,RUN"TEXT". Diese Angabe ist bei dem Betriebs-System NEWDOS möglich. Im Betriebs-System VTOS ist dieser Durchstart auch möglich, wird aber über ein Programm eingegeben. Ideal wäre es nun, sich dieser Systeme zu bedienen. Sie haben nur einen kleinen Fehler, sie sind verhältnismäßig teuer. Doch kann auch das TRS-DOS auf Durchstart umgestellt werden.

TRS-DOS-Erweiterung

Im DOS-Basic-Betrieb wird das ‚DEBUG'-Programm mit CMD"D aufgerufen. Es erscheint die übliche Auflistung im Hex-Dump *(Bild 1)*. Die letzten vier Zeilen geben den Inhalt der Speicher 0000 bis 0030 hexadezimal wieder. Nun tippen wir D6B00 ein und drücken Enter. Jetzt zeigen die letzten vier Zeilen den Inhalt der Speicher 6B00 bis 6B30 an. Dies wollen wir ändern.

Geben wir ein: M6B00 und Leertaste (Space), *nicht* ‚Enter'!! Links unten erscheint die Ziffer 6B00, und darunter 02- (oder auch eine andere Ziffer, die dann auch in der viertletzten Zeile direkt rechts neben 6B00= steht). Beginnen wir mit der Eingabe des gleich folgenden Programms in Maschinensprache. Zuerst die beiden Ziffern 21 eintippen. Sie erscheinen neben dem 02- zusammen als 02-21. Wieder Leertaste drücken, nicht Enter! Es erscheint nun 6B01 FF-. Tippen wir nun das gesamte Programm ein, immer 2 Ziffern und dann die Leertaste:

```
21 0F 6B 22 16 40 21 2C
6B 22 2A 6B C3 2D 40 E5
2A 2A 6B 7E FE 0A CA 1F
6B 23 22 2A 6B E1 C9 21
E3 03 22 16 40 3E 0D C3
1D 6B 20 20 1F 42 41 53
49 43 0D 0D 0D 52 55 4E
22 53 54 41 52 54 22 0A
```

Wenn wir die letzten beiden Ziffern eingegeben haben, steht links unten 6B3F FF-0A. Wir drücken noch einmal die Leertaste, und dann vergleichen wir noch einmal. Während der Eingabe konnten wir feststellen, daß auf dem Hex-Dump unten sich 2 gra-

```
AF = 44 42 -Z----N-
BC = 5B 01 =>  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
DE = 5B AD =>  44 00 00 00 44 00 0A FE  5B FE 2A 20 69 E5 2A B1
HL = 40 B7 =>  5B FF FF FF FF FF FF FF  FF 00 00 00 00 00 00 00
AF'= 00 44 -Z---P--
BC'= 4D 5C =>  5F C3 8E 59 C3 33 60 C3  7D 5B C3 8C 5B C3 A1 60
DE'= 01 08 =>  4F 52 59 20 53 49 5A 45  00 52 41 44 49 4F 20 53
HL'= 4D 00 =>  C3 46 5E C3 8E 55 C3 49  5E C3 55 56 C3 4C 5E C3
IX = 40 15 =>  01 78 43 00 00 00 4B 49  07 58 04 31 3E 00 44 4F
IY = 40 00 =>  C3 96 1C C3 78 1D C3 90  1C C3 D9 25 C3 A2 4B C3
SP = FF C7 =>  AE 5B 1E 1D 00 00 A3 45  A3 45 A3 45 A3 45 47 45
PC = 57 08 =>  E1 C9 3A 29 5B F6 C0 CD  09 44 E1 C9 D7 E5 3E 11
     6B00 =>  21 0F 6B 22 16 40 21 2C  6B 22 2A 6B C3 2D 40 E5
     6B10 =>  2A 2A 6B 7E FE 0A CA 1F  6B 23 22 2A 6B E1 C9 21
     6B20 =>  E3 03 22 16 40 3E 0D C3  1D 6B 3F 6B 1F 42 41 53
     6B30 =>  49 43 0D 0D 0D 52 55 4E  22 53 54 41 52 54 22 0A
```

Bild 1. Der Hex-Dump mit den letzten vier Zeilen der Speicherstellen 6B00 bis 6B3F, die hier schon geändert sind

fische Zeichen (senkrechte Striche sollen sie darstellen) um die beiden Ziffern legten, die auch links abgebildet wurden. Mit der Eingabe des Programms änderte sich dann auch die so gekennzeichnete Ziffer. Durch die Eingabe des Programms haben wir die gesamten 4 letzten Zeilen geändert, und können nun die Änderung noch einmal vergleichen. Stimmt alles, drücken wir Enter. Unser Eingabefeld links unten verschwindet wieder. Wir geben jetzt ein: G402D. Unsere Diskette springt an, es erscheint das bekannte DOS READY, wir sind wieder im DOS-Betrieb.

Der Start von Autostart

Unsere DOS-Diskette besitzt ein Programm TAPEDISK, das wir nun aufrufen:

TAPEDISK/CMD (Enter).

Die Diskette startet kurz an, und unter unserer eingetippten Anforderung erscheint:

?_.
Wir tippen nun ein:

F START/CMD:0 6B00 6B3F 6B00.

Hierbei ist unbedingt auf die Leerstellen zu achten! Werden diese vergessen, müssen wir noch einmal von vorn beginnen.

Nach dieser Eingabe folgt wieder Enter, und es erscheint wieder ein ?_, das wir nun mit E und Enter beantworten. Die Diskette startet wieder durch und antwortet mit DOS READY. Rufen wir nun mit DIR die DIRECTORY auf. *Bild 2* zeigt, wie sich unser Programm als START/CMD eingefügt hat. Wir können nun AUTO START/CMD eingeben, wie es *Bild 3* zeigt. Drücken wir anschließend den Reset-Knopf. Und nun passiert das, was wir uns schon immer wünschten: Die Diskette startet durch, über HOW MANY FILES und MEMORY SIZE bis zu RUN"START", und meldet dann... PROGRAMM NOT FOUND. Warum dieses? Nun, wir haben es hier mit zwei verschiedenen Programmen zu tun, die sich Start nennen. Da ist einmal START/CMD. Dieses Programm haben wir mit TAPEDISK in die DIRECTORY gesetzt. Doch in dem von uns eingegebenen Maschinenprogramm kommt noch einmal START vor. Betrachten wir uns dieses Programm etwas genauer, wenigstens die letzten 19 Bytes.

Auf dem Speicherplatz 6B2D (Bild 1) steht 42. Das ist das Hex-Zeichen für B. 41 ist A, 53 ist S, 49 ist I und 43 ist C. Betrachten wir diese Buchstaben im Zusammenhang, bedeutet das BASIC, und genau das ist ja der erste Aufruf, den wir zum Start ins Basic vom DOS her benötigen. Die 3 folgenden 0D bedeuten ‚Enter'. Erinnern wir uns, nach der Eingabe BASIC müssen wir auch erst einmal ‚Enter' drücken, dann noch einmal wenn HOW MANY FILES erscheint, was ja gleichzeitig bedeutet, es sind 3 FILES reserviert, und noch einmal ‚Enter' bei MEMORY SIZE, womit wir den höchsten Speicherplatz angeben. Es folgen dann die Hex-Zeichen 52 55 usw. bis 22. Und diese Zeichen bedeuten: RUN"START". 0A ist dann noch einmal Zeilenvorschub.

Dieses hier genannte Progamm START ist also ein Basic-Programm, das wir noch nicht haben. Die einfachste Methode ist nun, einfach im Basic-Modus einzugeben:

```
10 NEW
SAVE"START"
RUN
```

Bild 3 zeigt, wie dann auch dieses Programm in der DIRECTORY aufgeführt wird. Nach unserem AUTO-Start läuft nun die Diskette bis ins Basic durch (*Bild 4*). Unser Ziel ist erreicht.

Bild 2. Directory mit den beiden Programmen START/CMD und START

```
FILE DIRECTORY --- DRIVE 1      TRSDOS    -- 07/21/80

TEST1/CMD          GETDISK/BAS        START
TEST2/BAS          ZERO/CIM           DISKDUMP/BAS
GRAFIK             GETTAPE/BAS        TAPEDISK/CMD
START/CMD          RSM/CIM

DOS READY
```

Bild 3. Directory, fertig zum Auto-Start

```
FILE DIRECTORY --- DRIVE 1      TRSDOS    -- 07/21/80

TEST1/CMD          GETDISK/BAS        START
TEST2/BAS          ZERO/CIM           DISKDUMP/BAS
GRAFIK             GETTAPE/BAS        TAPEDISK/CMD
START/CMD          RSM/CIM

DOS READY
AUTO START/CMD_
```

Bild 4. Der Start ist bis ins Basic „durchgelaufen"

```
                DISK BASIC VERSION 2.0

      READY
      >
      >
      >RUN"START"
```

Tips zum Betrieb

Selbstverständlich kann statt START jeder andere Name gewählt werden – bei TAPEDISK jeder mit acht Buchstaben maximal, oder auch Zahlen, wobei die erste Stelle aber immer ein Buchstabe sein muß. Wollen wir in Basic ein anderes Programm aufrufen, müssen wir das gleich im Maschinenprogramm ändern. Dazu müssen wir die Speicherstellen 6B39 bis 6B3D umbenennen, und hier sind es nur fünf Buchstaben, die wir verwenden können. Haben wir also ein längeres Programm mit einem ebenso langen Namen, müssen wir diesen eben abkürzen, z. B. statt STARTREK eben nur STATR. Und diese Buchstaben geben wir mit ihren Hex-Zeichen in die genannten Speicherplätze ein. Eine weitere Möglichkeit wäre, das Programm START zu ändern. Statt

```
10 NEW
```

geben wir eine Übersicht der wirklich auf der Diskette befindlichen Programme an und rufen somit nach dem START ein Inhaltsverzeichnis auf.

Dr. Horst Spitschka, Horst W. Albrecht

Probleme des EDV-Unterrichts

Die Kultusministerien der Länder der Bundesrepublik Deutschland haben sich weitgehend darauf geeinigt, bis spätestens 1985 jede Schule mit einer ausreichenden Zahl von Computern für Ausbildungs- und Verwaltungszwecke auszurüsten. Wie die heutige Situation aussieht, lesen Sie im folgenden Beitrag.

Bevor wir uns der Situation des EDV-Unterrichts in den verschiedenen Schulbereichen zuwenden, wollen wir uns zunächst mit der Fortbildung von Erwachsenen befassen. Sie wird schon über eine Reihe von Jahren betrieben und dürfte Erkenntnisse vermitteln, welche sich auf den Schulunterricht übertragen lassen und wertvolle Ansätze für dessen praxisnahe Gestaltung bieten.

In der Erwachsenenbildung werden viele EDV-Kurse abgehalten, wobei vor allem die Hardwarehersteller und einige EDV-Schulen das Hauptgewicht tragen. Die Grundsätze dieser Kurse lauten in etwa:

- Viel Anwendung;
- relativ wenig Hardware;
- um so mehr Programmierung und
- den Teilnehmern die Angst vor dem Computer nehmen.

Ausgehend von der Tatsache, daß es sich bei den Teilnehmern um Erwachsene handelt, die lange nicht mehr die Schulbank gedrückt haben und daher das Lernen nicht mehr gewohnt sind, ergeben sich wichtige methodisch-didaktische Erkenntnisse, u. a. nämlich die Aktivität der Teilnehmer zu fördern.

Neben dem Lernwillen bringen die Kursteilnehmer oft viel praktisches Wissen und Erfahrungen mit, welche unseren Schülern wiederum fehlen. Entscheidend ist daher für unsere Schulen die Frage, was die Praxis später einmal von den Schülern erwartet. In den Erwachsenenkursen werden die Grundsätze „Anschaulichkeit" und „Selbsttätigkeit" in den Vordergrund gestellt. So sollte es dementsprechend auch in der Schule sein.

Situation an den Schulen

Im gymnasialen Bereich bietet man heute den Schülern das Fach Informatik an. In den Kollegstufen werden Grund- und Leistungskurse gehalten. Noch relativ bescheiden – wohl auch wegen der fehlenden Computer – ist der Einbau in den Unterricht (z. B. Mathematik, Physik, Wirtschaftslehre, Sozialkunde usw.) zu nennen.

Ein großes Hindernis für eine entsprechende Breitenentwicklung, allerdings nicht nur bei Gymnasien, ergibt sich aus folgenden Tatbeständen:

- Es sind keine – oder zu wenige – Computer vorhanden. Optimal wäre es, drei Schüler an einen Computer setzen zu können.
- Manche Lehrer, eben aus Mangel an ausreichender Hardware, verstehen Informatik als Fortsetzung des Mathematikunterrichts und vergraulen damit viele Schüler.
- Es gibt noch zu wenig methodisch aufgebaute Lehrbücher. Überhaupt fehlt noch eine Methodik des EDV-Unterrichts. Hier mag der globale Hinweis gelten: Soviel Schüleraktivität wie möglich!
- Die Ausbildung der Lehrer ist erst zögernd angelaufen, und vielen Kollegen aller Schularten ist der Computer immer noch eine Art „Schreckgespenst".

Die Realschulen bieten Informatik als Wahlfach an. Wie können jedoch Schüler dafür gewonnen werden, wenn nicht zumindest ein Schulcomputer vorhanden ist? Aber selbst dann würde auch wieder ein Problem entstehen: Ein EDV-Unterricht mit nur einem Gerät für eine größere Anzahl von Schülern ist praktisch wertlos. Dann sitzt nämlich ein Schüler am Terminal, einige weitere bekommen noch mit, was dort geschieht, während die anderen Schüler hinter dem Rücken des Lehrers blödeln oder herumstehen, weil es ihnen einfach zu langweilig wird. Da hilft auch die Übertragung auf einen Bildschirm sehr wenig.

Die Lösung wäre eine Aufteilung in kleine Guppen, was meist nicht genehmigt wird oder mangels genügender Computer nicht durchführbar ist. Diese Erfahrungen gelten allerdings nicht nur für die Realschulen allein. Diese Probleme sind die gleichen wie im gymnasialen Bereich, nämlich
- keine bzw. noch keine Hardware,
- keine geeigneten Bücher und
- nur wenige ausreichend ausgebildete Lehrer.

Ein allgemeiner Hinweis noch zu den Lehrplänen: Es gibt jetzt auch probeweise Lehrpläne. Jeder Lehrer, der Datenverarbeitung unterrichtet, sollte keine Gewissensbisse bekommen, wenn er quantitativ diese Lehrpläne nicht erfüllt. Qualität sollte auch hier vor Quantität gehen.

An den beruflichen Schulen wird das Fach Informatik, wie es an Gymnasien und Realschulen heißt, als „Datenverarbeitung" bezeichnet. Berufsoberschulen und Fachoberschulen haben mit ca. 80...90 Stunden EDV-Unterricht genügend Stunden zur Verfügung. An diesen Schulen dürfte auch die Ausstattung mit Hardware schon relativ gut sein.

Dagegen ist der Computer an anderen Berufsschulen noch selten zu finden. An diesen Schulen taucht dann neben den bereits erwähnten Problemen noch die Frage der zu geringen Stundenzahl

auf. Ein einstündiges Fach, d. h. ca. 15 Stunden Hardware und 15 Stunden Software dürfte auf Dauer zu neuen Überlegungen führen.

Die Wirtschaftsschulen haben in ihrem Fächerkanon „Organisationslehre" enthalten. Zusätzlich gibt es noch den Wahlunterricht in EDV. Die Probleme entsprechen exakt denen anderer Schulen. Noch sehr selten taucht bisher die Frage auf, was denn mit den Hauptschulen sei. Man braucht kein Prophet zu sein, wenn man unterstellt, daß auch diese Schulart eines Tages mit entsprechenden Anliegen an das Kultusministerium herantreten wird oder vielleicht sogar herantreten muß.

Da die Hauptschüler einen sehr erheblichen Teil der Schulabgänger und in vielen ihrer späteren Berufe mit der betrieblichen Praxis und dabei sicherlich auch mit der Datenverarbeitung (EDV) konfrontiert werden, ist es nur eine Frage der Zeit, daß entsprechende Aktivitäten zum Vorschein kommen.

Großes Interesse der Schüler

Neben dem bei nahezu allen Schularten vorhandenen mehr sachlichen Problemen gibt es aber noch viele ungelöste Probleme didaktischer Art:

1. Die Schüler bringen in der Regel großes Interesse für den EDV-Unterricht mit. Diese Begeisterung gilt es durch eine gezielte Schüleraktivität zu erhalten.
2. Das Programmieren sollte Vorrang haben. Das Wissen über die Hardware läßt sich aus den Fragen, die sich beim Programmieren zwangsläufig ergeben, behandeln.
3. Die richtige Vorgehensweise im Unterricht dürfte durchwegs induktiv sein, wobei möglichst viele Beispiele mit darauf aufbauenden Erklärungen zu behandeln sind.
4. Der Lehrer muß den Mut zur Stoffbegrenzung und zur Anpassung an die Erfordernisse der Schüler haben, d.h. er muß die Grenzen der für die Schüler relevanten Lehr- und Lerninhalte festlegen.
5. Es gibt in der EDV Lehr- und Lernhilfen, die der Lehrer allerdings für seine speziellen Zwecke zusammenstellen muß. Dazu gehört es, sich über das Angebot an Büchern, Programmen und AV-Lehrmitteln wie Unterrichtstransparenten, Tonbild-Reihen, Video-Programmen etc. laufend zu informieren.
6. Die Aufgaben sollten nicht zu umfangreich sein und aus den den Schülern bekannten Fachgebieten entnommen werden.

Wer versuchen wollte, Real-, Berufs- oder Wirtschaftsschülern z. B. Programme für Kubische Gleichungen, mit statistischen Meßzahlen oder schwierige physikalische oder chemische Formeln schreiben zu lassen, wird Schiffbruch erleiden.

Zusammenfassung

Es gibt heute noch keine Methodik und Didaktik des EDV- bzw. Informatikunterrichtes. Für Kollegen, die sich hier einen Namen machen wollen, sei damit eine Marktlücke aufgezeigt. Bei den verschiedenen Schularten schwankt die für den EDV-Unterricht vorgesehene Zeit zwischen 30 und 90 Unterrichtsstunden. Dabei sind die 30 Stunden (15 Stunden Hardware und 15 Software) der Berufsschulen sicherlich zu wenig.

Die Forderungen der Wirtschaft nach einem fundierten Grundwissen bei Schulabgängern aller Schularten werden es notwendig machen, den Schulunterricht quantitativ und qualitativ auf die Praxis auszurichten. Man sollte sich nicht darauf verlassen, daß Berufsanfänger später ja die Möglichkeit bekommen, sich im Rahmen der beruflichen Fortbildung dieses Wissen anzueignen. Nur wenige Unternehmen aus Industrie, Handwerk, Handel, Gewerbe und Dienstleistung können sich den im vorigen Aufsatz beschriebenen Ausbildungsaufwand leisten. Die meisten werden bei Einstellungsbewertungen vorhandenes EDV-Grundwissen als Qualifikationsmerkmal voraussetzen und danach ihre Personalentscheidung treffen.

Dem Schulbereich fällt damit die wichtige Aufgabe zu, alle notwendigen Voraussetzungen zu schaffen, um Schulabgängern – nicht zuletzt auch der Hauptschulen – dieses EDV-Grundwissen zu vermitteln.

Pseudo-Reset beim CBM 3001

Im Sonderheft der FUNKSCHAU Nr. 31, „Programme für Kleincomputer und Taschenrechner", veröffentlichte Stefan Burmeister ein Programm in Maschinensprache für das „alte" PET-Betriebssystem, das oftmals auch ohne speziell eingebaute Reset-Taste und vor allem ohne Programmlöschung einen „abgestürzten" Computer durch Druck auf die Stop-Taste aus dem Weltraum zurückholt. Im Bild ist eine an den CBM angepaßte Version abgedruckt, die alle nötigen Adressenkorrekturen enthält. *Harald Lindner*

```
An- oder Abschalten mit SYS826

033A 78        SEI            Verhindere Interrupt
033B A5 90     LDA 90         Vertauschung der Interrupt-
033D 49 67     EOR #67        vektoren E62E und 0349
033F 85 90     STA 90
0341 A5 91     LDA 91
0343 49 E5     EOR #E5
0345 85 91     STA 91
0347 58        CLI            Lasse neuen Interrupt zu
0348 60        RTS            Rueckkehr zu BASIC
0349 AD 27 80  LDA 8027       Invertiere rechtes
034C 49 80     EOR #80        oberes Bildschimfeld
034E 8D 27 80  STA 8027
0351 A5 98     LDA 98         Ist die Shift-Taste gedrueckt?
0353 F0 1D     BEQ 0372       Nein: -->
0355 A5 97     LDA 97         Ist die
0357 C9 1B     CMP #1B        Return-Taste gedrueckt?
0359 D0 17     BNE 0372       Nein: -->
035B BA        TSX            Lade X-Register mit Stackpointer
035C A9 C3     LDA #C3        Hi Byte Adresse BASIC-Warm-Start
035E 9D 06 01  STA 0106,X
0361 A9 89     LDA #89        Lo Byte
0363 9D 05 01  STA 0105,X     fuer neuen IRQ-Return speichern
0366 A9 08     LDA #08        Fuer eigene Tastaturabfrage
0368 8D 10 E8  STA E810
036B AD 12 E8  LDA E812       Ist
036E C9 DF     CMP #DF        Shift/Return noch gedrueckt?
0370 F0 F4     BEQ 0366       Ja: Tastatur nochmal abfragen
0372 4C 2E E6  JMP E62E       IRQ normal fortsetzen
```

Dieses Programm gestattet das „Zurückholen aus dem Weltraum" beim CBM. Es wird mit SYS 826 in Betrieb genommen und sorgt dann dafür, daß beim Druck auf Stop der Computer wieder ansprechbar ist – vorausgesetzt, der Interrupt-Vektor wurde nicht geändert

Hans-Georg Joepgen

Mehr Komfort bei Apple-Disks

Das unlängst auf dem Markt erschienene neue Disketten-Betriebsprogramm DOS 3.3 für die Computer ITT-2020 und Apple-II zeichnet sich gegenüber Vorläuferversionen und manchen Konkurrenzprodukten durch eine Fülle von Vorzügen aus, zu denen vor allem eine eklatante Erhöhung der Schreibdichte gehört. Der folgende Beitrag berichtet über erste Betriebserfahrungen mit DOS 3.3 und über Kompatibilitätsuntersuchungen.

Bereits die Vorgänger-Versionen von DOS 3.3 wiesen, verglichen mit handelsüblichen Disketten-Betriebssystemen verbreiteter Konkurrenz-Rechner, zahlreiche Eigenheiten auf, die man andernorts noch heute vermißt: Bequeme Errichtung von Textaufzeichnungen wahlweise für sequentiellen oder wahlfreien Zugriff [1], Abspeicherung von Programmen entweder zur besseren Speicherausnutzung in vorkompiliertem Zwischencode oder zur bequemen Editierung als ASCII-Text [2] und schließlich die Möglichkeiten, im Bedarfsfall nichtresidente Hochsprachen automatisch laden zu können und den Rechner über die Betriebsart EXEC (Execute) durch Textfiles zu steuern. Die DOS-Varianten bis zur Version 3.21 einschließlich formatierten ihre Fünf-Zoll-Disketten in 35 Abtast-Kreisen (Tracks) zu je 13 Abschnitten (Sectors), wobei jeder dieser Sektoren Raum für 255 Nutz-Bytes bietet. Vier der Abtast-Kreise enthalten das DOS selbst sowie den von DOS zur Aufzeichnungsverwaltung benötigten Speicherraum, so daß für den Benutzer gegen 100 KByte übrigblieben.

Neue PROMs machen's möglich

DOS 3.3 bringt auf der gleichen Mini-Diskettenfläche pro Teilkreis 16 Abschnitte unter, wodurch das Netto-Volumen auf fast 124 KByte steigt: Eine Kapazitätserweiterung um nahezu ein Viertel!

Damit nach der Einschaltung des Rechners das Disk Operating System überhaupt in den Computer geladen werden kann, muß eine Minimalausrüstung an Lese- und Positionierroutinen im Mikro-Computer resident vorhanden sein. Und wie bekannt, ist derlei von Haus aus in den Maschinen der Apple-Klasse nicht anzutreffen. Jene Geburtshilfe-Programmsegmente, die jeweils den rund 10 KByte von DOS das Überwechseln von der Initialisierungsdiskette in den RAM-Bereich des Rechners ermöglichen, sind vielmehr auf dem „Disk Controller" in zwei PROMs untergebracht – auf jener Zusatzplatine, die in einem der Computer-Steckplätze montiert wird und ihrerseits dem Anschluß von bis zu zwei Diskettenstationen dient. Diese zwei PROMs gilt es auszuwechseln; sie liegen den im Fachhandel ab sofort erhältlichen Umrüstsätzen einschließlich Werkzeug bei. Weiter enthält der Umrüstsatz zwei Disketten, einen „System Master" mit DOS 3.3 selbst und eine BASICS genannte Scheibe, die über zwei höchst wünschenswerte Eigenschaften verfügt: Sie dient einmal der Initialisierung einer eventuell vorhandenen Erweiterungskarte „Language Card", die das UCSD-Pascal-Betriebssystem der Apple-Maschinen beherbergt, auf Basic-Betrieb. Zum zweiten wird trotz neuer PROMs nach dem „Booten" von BASICS DOS 3.3 in die Lage versetzt, im alten 13-Sektor-Betrieb früherer DOS-Versionen zu arbeiten: Wichtig zur Aufrechterhaltung der Kompatibilität mit älterer Software – darüber jedoch unten weiteres.

```
]CATALOG
DISK VOLUME 254
*A 006 HELLO
*I 018 ANIMALS
*T 003 APPLE PROMS
*I 006 APPLESOFT
*I 026 APPLEVISION
*I 017 BIORHYTHM
*B 010 BOOT13
*A 006 BRIAN'S THEME
*B 003 CHAIN
*I 009 COLOR DEMO
*A 009 COLOR DEMOSOFT
*I 009 COPY
*B 003 COPY.OBJ0
*A 009 COPYA
*A 010 EXEC DEMO
*B 020 FID
*B 050 FPBASIC
*B 050 INTBASIC
*A 028 LITTLE BRICK OUT
*A 003 MAKE TEXT
*B 009 MASTER CREATE
*B 027 MUFFIN
*A 051 PHONE LIST
*A 010 RANDOM
*A 013 RENUMBER
*A 039 RENUMBER INSTRUCTIONS
*A 003 RETRIEVE TEXT
```

Das neue Disketten-Betriebssystem DOS 3.3 für die Maschinen der Apple-Klasse bringt nicht nur eine Erhöhung der Schreibdichte, sondern auch vielerlei zusätzliche Nutzroutinen – hier das Inhaltsverzeichnis der „System Master Diskette"

Dienstroutinen schaffen Flexibilität

Schließlich erhält man beim Erstehen des Umrüstsatzes noch ein ausgezeichnetes Handbuch in hervorragendem Druck auf strapazierfähigem Papier – knapp 200 Seiten didaktisch und typografisch vorbildlich gestaltete Tips für den Umgang mit DOS 3.3, mit Tabellen und Demonstrationsprogrammen, alles in unkompliziert formuliertem Englisch.

Die Firma Apple Computer Incorporated in Cupertino ist bekannt dafür, daß sie den Systemdisketten ihrer bisheri-

gen DOS-Versionen regelmäßig zahlreiche Dienstroutinen und Demonstrationsprogramme beipackte, zu denen bisher schon ein höchst komfortabel gestaltetes Renumerierungsprogramm gehörte. Die Tradition wird mit der DOS-3.3-Systemdiskette fortgesetzt. Bemerkenswert sind hier vor allem die in Maschinencode vorliegenden Programme „MUFFIN" und „FID". MUFFIN liest 13-Sektor-Disketten und schreibt im neuen 16-Sektor-Format auf Disketten zurück: Und zwar entweder den gesamten Disketten-Inhalt oder aber ausgewählte Aufzeichnungen. Diese schöne Möglichkeit steht nicht nur den Besitzern von mit mindestens zwei Laufwerken ausgerüsteten Systemen offen, sondern auch dem mit lediglich einer Diskettenstation versehenen Benutzer. Hier sagt MUFFIN dem Mann am Rechner, wann er jeweils die 13-Sektor-Diskette und wann er die 16-Sektor-Diskette einzuführen hat. Die im alten Format hergestellte Scheibe bleibt bei all dem unverändert, es wird lediglich eine Kopie im 16-Sektor-Format gefertigt.

DOS 3.3 besser als CP/M?

Vieles Vermißte stellt nun der „File Developper" FID zur Verfügung: Kopieren jetzt auch von Textfiles und Binäraufzeichnungen mit gruppenweiser Behandlung, blockweises Sichern, Löschen und Entsichern, transparente Buchführung über die Anzahl der belegten und verfügbaren Netto-Nutz-Sektoren auf Disketten – all dieses wieder wahlweise im Ein-Laufwerk-Betrieb oder mit mehreren Disketten-Stationen und durch narrensichere Soufflier-Tafeln mit hohem Bedienungskomfort ausgestattet. Die von Drittlieferanten angebotenen Disketten-Betriebssysteme CP/A und Apex zeigen zwar ein ähnliches Leistungsvermögen, doch ist die Bedienung hier wesentlich komplizierter, mancherlei Restriktionen (Präfix bei Aufzeichnungsnamen maximal acht Zeichen, Suffix maximal drei Zeichen) engen hierbei den Programmierer-Spielraum ein. Dies gilt selbst für das berühmte Disketten-Betriebssystem CP/M, das seit neuestem ebenfalls für die Maschinen der Apple-Klasse zur Verfügung steht (freilich erst nach einer Hardware-Erweiterung: Einbau der „Softcard-Platine" von der Firma Microsoft – Consumer Products. Diese Steckkarte enthält eine eigene CPU Z80). Das Bild zeigt den Katalog der System-Master-Diskette von DOS 3.3, Inhalt:

Außer zahlreichen Spielen, Demonstrationensprogrammen und Nutzroutinen einschließlich der bereits erwähnten FID, MUFFIN und Renumerierungsroutine enthält die Diskette Aufzeichnungen einer „COPY"-Gruppe, die zur schnellen Diskettenkopierung auf unformatierte Scheiben dient. „MASTER CREATE" verwandelt existierende Satelliten-Disketten („Slaves") in eigenständige Master-Scheiben, ohne bereits vorhandenen Benutzer-Inhalt dabei zu verändern. Die Unterschiede zwischen DOS 3.3 und seinen Vorgängerversionen sowie die weiterhin fortbestehenden Gemeinsamkeiten zeigt in Form einer Übersicht die Tabelle.

Tabelle: Vergleich der „alten" und „neuen" DOS-Version

	DOS 3.2	DOS 3.3
Abtast-Kreise (Tracks)	35	35
Kreis-Abschnitte (Sectors)	13	16
Brutto-Kapazität (Bytes)	116480	143 360
Netto-Kapazität (Bytes)	103168	126 976
Beginn Directory (Track, Sector)	$11,$0C	$11,$0F
Vom System selbst belegt (Tracks)	$0-$2, $11	$0-$2, $11

Kompatibilität voll gewahrt

Da gibt es das unrühmliche Beispiel eines auch auf dem deutschen Markt mit hohen Stückzahlen vertretenen Mikrocomputer-Herstellers, der bei der Fortentwicklung seiner Hard- und Betriebssoftware auf die Käufer der Vorversionen in der Vergangenheit herzlich wenig Rücksicht nahm. Den Apple-Managern in Cupertino und ihren Kollegen bei Standard Elektrik Lorenz in Pforzheim (ITT) ist eine derlei anfechtbare Geschäftsmoral augenscheinlich fremd; wie man hört, wurde auf so manche recht erfolgversprechende Innovation so lange verzichtet, bis die Verträglichkeit der Neuerung mit zuvor verkauften Geräten und Programmen sichergestellt war.

Dies gilt auch für DOS 3.3; durch die Fähigkeit des neuen Disketten-Betriebssystems, voll in das Kleid seiner Vorgängerversionen zu schlüpfen, ist absolute Weiterverwendbarkeit früher geschriebener Programme gesichert. Erforderlich ist dieses Umschalten auf Alt-DOS *(Tabelle)* freilich nur in sehr wenigen Fällen: Dann nämlich, wenn gewisse hausfremde Betriebssysteme oder Programme gefahren werden sollen, zum Beispiel Focal, Forth, XPLO, CP/A oder Apex.

Bei unseren Kompatibilitätsuntersuchungen kam folgendes heraus: Immer dann, wenn von einer Standard-Sprache mit Standard-DOS-Mitteln auf die Scheiben zugegriffen wird, läuft die mit MUFFIN konvertierte Software problemlos im 16-Sektor-Format. Nur dann, wenn die Ursprungsprogramme eigenständige DOS-Erweiterungen enthalten, sollte man auf 13-Sektor-Betrieb zurückgehen oder aber die Assembler-Passagen einer solchen Software modifizieren.

DOS-Erweiterungen anpassen

Man sucht mit einem geeigneten Disassembler oder einem Feldanalyse-Programm, wie in [3] vorgestellt, jene Programmstelle auf, von der die Unterroutine RWTS („Read or Write a Track and Sector", JSR $ 03D9) gerufen wird.

Zum Zeitpunkt des Aufrufs enthalten interne Register der CPU die Adresse eines „IN/OUT CONTROL BLOCK" (höherwertiges Byte im Akkumulator, niederwertiges Byte im Y-Register). Das sechste Byte dieses Blocks enthält die Nummer des Sektors, auf den RWTS jeweils zugreift. Beim Absuchen des Disketten-Inhaltsverzeichnisses („DIRECTORY") beginnen derlei Programme üblicherweise hier mit dem Wert $0C; es folgen $0B, $0A, $09 und so fort. Zur Adaption an den 16-Sektor-Betrieb ist es nun erforderlich, dafür Sorge zu tragen, daß der Startwert im Byte 6 des I/O-Blocks $0F beträgt, sonst werden die zusätzlichen Sektoren vom System „übersehen".

Beim Assembler „ASMB 6502" der Firma Digital Marketing und dem „Text Processing System" des gleichen Unternehmens stießen wir bei der Erprobung von DOS 3.3 auf den merkwürdigen Fehler, daß Textsystem und Assembler die vom eigenen Editor auf Diskette deponierten Textsammlungen bisweilen nicht fand – und zwar, wie sich nach einigen Stunden Analysierens herausstellte, immer dann, wenn der Directory-Eintrag der betreffenden Files auf den Sektoren 13, 14 oder 15 erfolgt war. Die Änderung je eines einzigen Bytes in Assembler ($0A98, File „ASMBMACHCODE") und Textsystem ($0BA5, File „TEXTMACHCODE") von Wert 0C in den Wert 0F sorgte dafür, daß auch diesen weitverbreiteten Nutzprogrammen DOS 3.3 voll erschlossen wurde. Überflüssig zu sagen, daß diese Änderung lediglich für

16-Sektor-Betrieb erforderlich war, Inkompatibilität mit 13-Sektor-Betrieb traten hier, wie auch in keinem weiteren Fall, bei unserer Erprobung nicht auf.

Fazit

Schon bisher boten die Apple-Disketten-Betriebssysteme dem Benutzer mehr Netto-Kapazität auf seiner 5-Zoll-Scheibe als viele Konkurrenzprodukte. Die neuerliche Vergrößerung des Fassungsvermögens um knapp ein Viertel hat weiterhin nicht zu einer Verringerung der Betriebssicherheit geführt, wie sie auf manchen „Double-Density-Systemen" anderer Rechner beobachtet wurden. Zwar ist der Umrüstsatz auf DOS 3.3 (in Deutschland) arg teuer, doch rechtfertigen allein die durch den „File Developper" FID zusätzlich eröffneten Möglichkeiten in gewissen Fällen das Ausgeben einer immerhin dreistelligen Summe durchaus.

Übrigens: Wer sich in diesen Tagen seinen Apple II beziehungsweise die Disketten-Anlage dazu erst zu erwerben gedenkt, der achte darauf, die bereits von Haus aus mit DOS 3.3 ausgerüstete Version zu bekommen: Wie man hört, geistern noch frühere Varianten durch europäische Läger, und in einem Fall wurde bekannt, daß ein südwestdeutscher Händler den Versuch machte, durch listiges Verschweigen der Existenz von 3.3 offenbar erst mal seine Ladenhüter mit DOS 3.21 auf den Weg zum Kunden zu bringen. Also: Aufgepaßt, die Welt ist schlecht! Eine erfreuliche Nachricht zu guter Letzt noch für Besitzer der Apple-Language-Card: Sie brauchen keine neuen PROMs; die zum Pascal-System gehörenden Controller-ICs P5A und P6A sind in der Lage, DOS 3.3 zu laden [4] und zu treiben.

Literatur

[1] Valentini, Johannes: Ein (fast) ideales Speichermedium – Das Disk-Operating-System des Euro-Apple. FUNKSCHAU 1980, Heft 6, Seite 105–107. Franzis, München.

[2] Joepgen, Hans-Georg: „Compreter" und „Interpiler" auf dem Vormarsch. FUNKSCHAU 1980, Heft 14, S. 85–86. Franzis, München.

[3] Joepgen, Hans-Georg: Suchprogramm – formuliert in drei Sprachen. Hobbycomputer 2, Sonderheft der FUNKSCHAU, S. 27–28. Franzis, München.

[4] Joepgen, Hans-Georg: Pascal auf einem Mikrocomputer – Das UCSD-System des Apple. FUNKSCHAU 1980, Heft 12, S. 91–93. Franzis, München.

E-12-Reihe per Basic

Die folgende Routine kann in Programmen Verwendung finden, die zur Schaltungsdimensionierung dienen. Berechnete Werte (z. B. Widerstände oder Kapazitäten) sind meist „krumm", reale Bauelemente sind jedoch nur in fest abgestuften Werten erhältlich, z. B. in der Reihe E12. Die Routine besteht aus einem Test-Hauptprogramm (Zeilen 50...100) und der eigentlichen Konversionsfunktion (Zeilen 140...270).

Die Funktion benötigt als Parameter den „krummen" Eingangswert X. Zurückgegeben wird der E-12-Wert, der dem Eingangswert am nähesten liegt. Zusätzlich werden noch verschiedene Variablen gesetzt, die auch als Rückgabe-Werte verwendet werden können:

X2 = nähester E-12-Wert, der größer oder gleich dem Eingangswert ist
X3 = nähester E-12-Wert, der kleiner oder gleich dem Eingangswert ist
X4 = Abweichung (E-12-Wert minus Eingangswert)

X2 oder X3 können in den Fällen verwendet werden, wo X ein Minimal- bzw. Maximal-Wert ist und keine Abweichung in der falschen Richtung zulässig ist.

Die E-12-Grundwerte sind in Form einer DATA-Tabelle definiert (Zeile 290). Hier können auch die Grundwerte der Reihen E 6, E 24, E 48 usw. eingesetzt werden, je nach zulässiger Toleranz, oder eine beliebige, monoton steigende Zahlenfolge mit beliebig vielen Elementen. Wichtig ist nur, daß die DATA-Tabelle mit 1 beginnt und mit 10 endet. Die Konversion bildet zuerst den Dekadenbereich aus dem Eingangswert X (Zeile 150):
INT(lg X) = INT(lg e · ln X)
= INT(0.4342 · LOG[X])
Das ergibt den Exponenten des Dekadenbereichs; 10 hoch diese Zahl ergibt den Grundbereich (0.1, 1, 10, 100 usw.), in dem der Eingangswert X liegt. Die aus der DATA-Tabelle gelesenen E-12-Grundwerte werden jeweils mit diesem Grundbereich multipliziert (Zeile 200) und dann mit dem Eingangswert verglichen (Zeile 210). In der Variablen X3 wird jeweils der vorhergehende E-12-Wert zwischengespeichert (Zeile 180).
Sobald ein gelesener und angepaßter E-12-Wert größer oder gleich dem Eingangswert X ist, wird geprüft, ob der eben gelesene (X2) oder der vorhergehende (X3) E-12-Wert dem Eingangswert näher liegt. Dieser E-12-Wert wird dann der Variablen X5 zugewiesen (Zeilen 220 und 230) und mittels RETURN als Funktionswert zurückgegeben (Zeile 260). *Günter Ruschitzka*

```
50 PRINT "UMWANDLUNG VON EINGANGSWERTEN IN E12-WERTE"
60 PRINT
70 INPUT "WERT:       ",X
80 PRINT "E12-WERT: ",FNR(X)
90 PRINT \ PRINT
100 GOTO 70
110 REM
120 REM
130 REM ROUTINE ZUR KONVERSION IN E12-WERTE
140 DEF FNR(X)
150 X1=10^INT(.4342945*LOG(X)) \ REM DEKADENBEREICH BILDEN
160 RESTORE 290
170 X2=X1     \ REM ERSTER E12-WERT
180 X3=X2     \ REM ZWISCHENSPEICHERN DES VORHERGEHENDEN WERTS
190 READ X2   \ REM E12-WERT AUS TABELLE LESEN
200 X2=X2*X1 \ REM BEREICHSANPASSUNG
210 IF X>X2 THEN GOTO 180   \ REM NAECHSTEN E12-WERT LESEN
220 X4=(X2+X3)/2            \ REM MITTELWERT DER E12-WERTE
230 IF X>X4 THEN X5=X2 ELSE X5=X3 \ REM E12-WERT ZUORDNEN
240 IF X=X2 THEN X3=X2      \ REM INTERVALL IST NULL
250 X4=X5-X \  REM ABWEICHUNG
260 RETURN X5
270 FNEND
280 REM TABELLE DER E12-GRUNDWERTE
290 DATA 1,1.2,1.5,1.8,2.2,2.7,3.3,3.9,4.7,5.6,6.8,8.2,10
300 END
```

Dieses Basic-Programm verwendet einen auf kleineren Tischrechnern etwas unüblichen „Dialekt". Für Geräte wie PET, CBM, PC-100 usw. sind nach links geneigte Schrägstriche durch Doppelpunkte zu ersetzen, „X5" nach RETURN ist wegzulassen, und die Zeilen 270 und 300 können entfallen

COMMODORE OHIO-SCIENTIFIC VIDEO-GENIE

Superboard II 4K 795.–
Superboard II 8 K 870.–

FS-Adapter 5V	28.–
Challenger C1P 8K	ca. 1450.–
Challenger C4P 8K	1950.–
Floppy Superb. 80K	1450.–
Karte 610 bis 24K	850.–
Monitor 9Z, weiß	395.–
Monitor 12Z, grün	560.–
Discette 5Z Soft, 1 St.	9.95
Discette 5Z Soft, 10 St.	79.–
C 20 Cassette, 1 St.	1.95
C 20 Cassette, 10 St.	17.50
2114-2	9.50
4116-3	9.50
2716	17.50
2532	46.50
Gehäuse für Superboard	139.50
Netzteilkarte 5V/5A	89.–

Lieferung per NN. Preisliste frei. Informationspaket gegen 2 DM in Briefmarken. Preisänderungen vorbehalten. Video-Genie & Commodore führen wir nur in Köln.

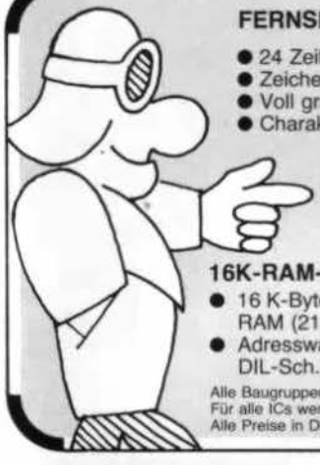
Video Genie EG 3003	1395.–
Expansion 32 K	1645.–
MX-80 Drucker	1695.–
MX-80 FT-Drucker	1895.–
Papier 2000 Bl. weiß	58.–
Interface MX-EG m. Kabel	190.–
Interface CBM-MX	a. Anfr.
Interface MX-Appel, m. K.	175.–
Interface MX-TRS 80, m. K.	239.–
Commodore 4016	2495.–
Commodore 4032	2923.–
Commodore 8032	3864.–
Commodore 4040	2923.–
CN-2 CBM-Recorder	265.–
Kabel CBM-Drucker etc.	115.–
Kabel CBM-PER-PER	145.–
Hameg Sonderpreise	a. Anfr.

5000 Köln 1, Aachenerstr. 27
Telefon (02 21) 23 79 08

Radio P+M Elektronik

5300 Bonn 1, Sternstr. 102
Telefon (02 28) 65 60 05

5000 Köln 80, Buchheimerstr. 23
Telefon (02 21) 61 71 61



FERNSEHINTERFACE CRT 1 (CHIP 7/8-80)

- 24 Zeilen à 32 Zeichen
- Zeichenmatrix 8 × 8
- Voll graphikfähig
- Charaktergenerator EPROM 2708

Platine + Handb.	89,–
Teilbausatz	196,–
Komplettbausatz	298,–
Fertiggerät	469,–
Nur Handbuch	20,–

FERNSEHINTERFACE CRT 2 (z.B. für AIM 65)

- 16 Zeilen à 64 Zeichen
- Zeichenmatrix 8 × 16
- Voll graphikfähig
- Charaktergenerator EPROM 2716

Platine + Handb.	89,–
Bausatz	398,–
Fertiggerät	569,–
Nur Handbuch	20,–

BAUSÄTZE FÜR MIKROCOMPUTER
GRAF ELEKTRONIK SYSTEME GMBH

Postfach 16 10 · 8960 KEMPTEN · TEL. (08 31) 6 19 30

SCHNELLDRUCKER EPSON MX 80
ORIGINALAUSDRUCK + SONDERPREIS
AUF ANFRAGE!

16K-RAM-PLATINE RAM 16 (CHIP 1/81)

- 16 K-Byte stat. RAM (21L14)
- Adresswahl über DIL-Sch.

Platine + Handb.	89,–
Bausatz mit 1-K-Byte-RAM	268,–
* RAMs (21L14)	9,90

ROM – PLATINE PROM

- 8/16/32 KByte
- Für EPROMS 2708, 2716, 2758 oder 2732
- Adressen über DIL-Sch.

Platine + Handb.	89,–
Bausatz (ohne EPROM)	169,–
Fertiggerät	279,–

Alle Baugruppen sind Europakarten. Alle Bausätze mit Markenhalbleitern, alle Platinen durchkontaktiert und mit Lötstoplack.
Für alle ICs werden Sockel geliefert!
Alle Preise in DM inkl. MwSt. ab Kempten. Angebote freibleibend. Umfangreiche Info kostenlos. Händleranfragen willkommen.

ges

Friktion/ Traktorführung kombiniert
Für Einzelblatt-, Rollen- und Endlospapier

EPSON MX 80 F/T

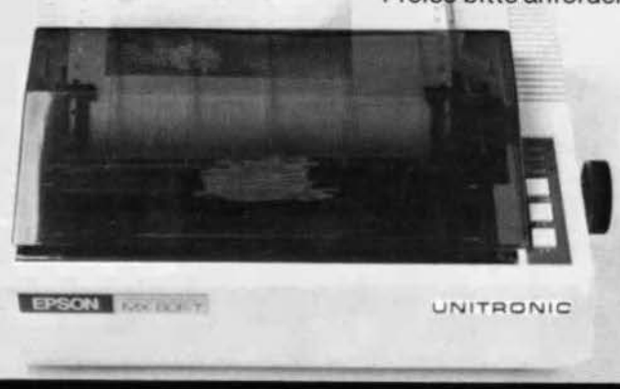
DM 1994,–
Interface ab DM 180,–
Kabel ab DM 95,–
Preise incl. MWSt. ab Lager
*Händler erhalten Original-WV-Konditionen

modernster Universaldrucker für alle Computer - Einzelblatt-, Rollen- u. Endlospapier - anschlußfertig mit jeweiligem Interface

MX 80 F/T, Microprozessorgesteuert mit 9x9 Matrix, Groß-Kleinschrift mit Unterlängen, 96 ASCII-Zeichen u. 64 Graphic und 8 internat. Zeichen (Umlaute, etc. umschaltbar), programmierbare Zeichen 40, 66, 80 oder 132 Zeichen/Zeile, bidirektionaler Druck mit Druckwegoptimierung, stufenlos verstellbare Papierführung bis 10", Normalpapier mit 4 Durchschlägen, Fettdruck, Tabulator, Lebensdauer min. 100.000.000 Zeichen, sehr geräuscharm - 57 dB, geringes Gewicht - 5 kg.

APPLE ITT 2020 — Interface 8131 Kabel 8230
PET — Interface 8110 Kabel 8210
TRS 80 — Interface 8120 Kabel 8220 BUS-Kabel 8221
Ti 99/4 — über RS 232 Interface 8140 Kabel 8240

Neu! Hochauflösende Grafik (Plotter) durch ROM-Nachrüstsatz
Preise bitte anfordern!



Delphin — Kabel 82 UD
NANO (SGS) — Interface 8140 Kabel 8240 über RS 232
Sharp MZ 80 — Interface 8180 Kabel 8280
Heath/Zenith WH 19 + WH 89 Wavemate IS 1000/S 2000 — Interface 8140 Kabel 8240 über RS 232

und ABC 80 - DAI - SORCERER - Compucorp - HP - IEE 488 - und, und, und . . .

UNITRONIC HAMBURG GMBH & CO KG
Lindhofstraße 3, 2360 Bad Segeberg
Telefon 04551 / 8697 + 8698
Telex 261646

UNITRONIC VERTRIEBS GMBH
Manskestraße 29, 3160 Lehrte
Telefon 05132 / 53001
Telex 922084

Epson-Vertragshändler
Computer+Components
Abt. UNITRONIC

UNITRONIC R. BREIDEN GMBH
Postfach 330429, Münsterstraße 338
4000 Düsseldorf 30
Telefon 0211 / 626364-67
Telex 0-8586434

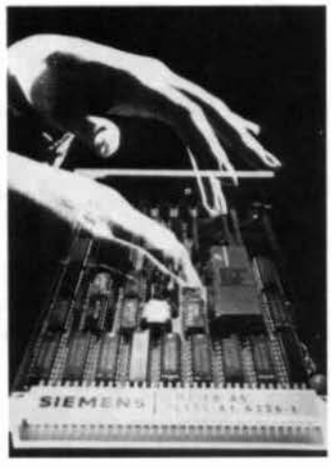
UNITRONIC GEORG GMBH & CO KG
Hochfelln 4
8019 Ebersberg
Telefon 08092 / 21333

Alphatronic in neuem Gewand

Nachdem Triumph-Adler erst kürzlich einen Großauftrag für ihr Alphatronic-Computersystem an die Deutsche Bundespost im Ausbildungsbereich verbuchen konnte, wird der Computer nun in Deutschland in Großserie gefertigt.
Die Geräte erhielten ein neues Aussehen, aber auch ein verbessertes Innenleben bezüglich Hardware und Software (Betriebssystem). Die alphanumerische DIN-Schreibmaschinen-Tastatur mit deutschen Umlauten und der Norm-Zehnerblock heben sich farblich von den Steuertasten (sechs programmierbare Funktions-Tasten, „soft keys" und fünf Cursortasten) ab. Eine neue SM-Umschalttaste schafft Wahlmöglichkeiten zwischen Eingaben in Schreibmaschinen-Art (Kleinschreibung bei nicht gedrückter Shift-Taste) und EDV-Funktion (Großschreibung ohne, Kleinschreibung mit Shift-Taste).
Im Gehäuse befinden sich wahlweise auch ein oder zwei Mini-Diskettenlaufwerke (160 KByte pro Floppy). Als Mikroprozessor dient der bekannte 8085; der Basic-Interpreter des „Alphatronic" ist allgemein als sehr komfortabel anerkannt.
Für selbstprogrammierende Anwender ist ein programmierter Basic-Lehrgang zu haben. Die derzeitige Programmbibliothek umfaßt u. a. Datei- und Adressenverwaltung sowie Zins- und Prognoseberechnungen.
Das derzeitige Alphatronic-System umfaßt die Versionen P1 (32 KByte RAM, 1 Disk-Laufwerk) und P2 (48 KByte RAM, 2 Laufwerke); in beiden Fällen wird der von Sanyo gefertigte Monitor mit grünleuchtendem Bildschirm und 12 Zoll Diagonale mitgeliefert, auf dem 24 Zeilen mit je 80 Zeichen dargestellt werden. Seit neuestem ist auch ein kompakter Nadeldrucker namens DRH 80 lieferbar, bei dem es sich um eine Triumph-Adler-Eigenentwicklung handelt.
(Triumph-Adler, Fürther Straße 212, 8500 Nürnberg)

SMP-System mit 8088

Für sein modulares Mikrocomputer-Baugruppen-System SMP bringt Siemens jetzt einen neuen Zentralcomputer mit dem vom 8086 abgeleiteten Prozessor SAB 8088 heraus; die bisher verfügbaren SMP-Baugruppen sind entweder mit dem 8080 oder mit dem 8085 bestückt. Die neue Baugruppe SMP-E8-A5 umfaßt Zentraleinheit, Speicher sowie Steuerung für direkten Speicherzugriff (DMA) und erschließt das SMP-System für den 16-Bit-Bereich. Der SAB 8088 arbeitet mit 5 MHz Taktfrequenz und benutzt den Befehlssatz des SAB 8086. Ge-



genüber den bisherigen Zentraleinheiten mit SAB 8080/85-Prozessoren kann der Datendurchsatz durch Stringbefehle erheblich gesteigert werden.
Der neue Zentralcomputer SMP-E8-A5 verfügt über 640 KByte Adreßraum, der in zehn Segmente zu je 64 KByte unterteilt ist. Die Segmente lassen sich durch Speicherbankumschaltung selektieren. Der Speicher auf dem Zentralcomputer umfaßt 4 KByte RAM (statisch) und zwei Fassungen für EPROM-Bausteine (SAB 2716, 2732 und 2764). Die DMA-Steuerung arbeitet mit 2,5 MHz Taktfrequenz, bietet vier Kanäle und erlaubt innerhalb der untersten 64 KByte des Adreßraums Datenübertragung sowohl zwischen verschiedenen Bereichen des Speichers als auch zwischen Speicher und Peripherie. Eine 96polige Messerleiste verbindet den Zentralcomputer mit dem SMP-Bus.
Für den Zentralcomputer SMP-E8-A5 liefert Siemens auch ein Monitorprogramm (SMP-MON8-A2), mit dem sich Anwenderprogramme ein- und ausgeben, anzeigen und verändern sowie im Dialog testen lassen. Um Programme ein- und ausgeben zu können, ist folgende Konfiguration erforderlich: Zentralcomputer SMP-E8-A5, serielle Ein-/Ausgabe-Baugruppe SMP-E220 und eine Datensichtstation. Das Monitorprogramm ist auf einer Floppy-Disk gespeichert und hat einen Umfang von 8 KByte.
(Siemens, ZI, Postfach 103, 8000 München 1)

Floppy für ABC-80

Die schwedische Firma Luxor, bekannt auch durch ihren Basic-Tischcomputer ABC-80, hat passend zu diesem Gerät ein Doppel-Floppy-Laufwerk mit 2 × 160 KByte Kapazität (5¼ Zoll) herausgebracht. Beide Laufwerke können individuell auf Single Density umgeschaltet werden, z. B. um frühere Aufzeichnungen zu übernehmen. Ein PLL-Datenseparator sorgt für eine einwandfreie Synchronisation auch bei leicht verschmutzten oder abgenutzten Floppies. Zur Stromversorgung ist ein verlustleistungsarmes Schaltnetzteil eingebaut. Ferner sind Steckplätze für maximal sieben Europakarten vorhanden, z. B. IEC-Bus-Interface, Relaisausgänge oder A/D-Wandler. Hierfür können die Karten der ebenfalls schwedischen Firma Databoard verwendet werden, die auch den ABC-80 für Luxor entwickelte.
Die Schweizer Firma Datormark hat übrigens den Vertrieb des ABC-80-Systems für den deutschsprachigen Raum übernommen, nachdem es bisher gewisse Exportschwierigkeiten gab.
(Datormark AG, Postfach 1130, CH-6301 Zug, Schweiz)

Programme für die Schulverwaltung

Für die Computerserien CBM 2000, CBM 3000, CBM 4000 und CBM 8000 brachte Horst W. Albrecht mehrere Schulverwaltungsprogramme heraus, die auf Kassette oder auf Floppy-Disk erhältlich sind.
Ein Programm für die Stundenplanerstellung (D/6-F) findet sich darunter ebenso wie Programme zur Schülererfassung und zum Erstellen von Statistiken (SVP-F, ZEP-F). Für den Fachunterricht stehen weiterhin mehrere Programme auf Kassetten zur Verfügung (Kostenrechnung, Rechnungswesen, Betriebswirtschaftslehre usw.).
(Horst W. Albrecht, Postfach 22, 8500 Nürnberg-Katzwang)

Wir fangen an, wo andere aufhören.

Beratung · Auswahl · Software
Service · Zubehör · Peripherie · Literatur

Die Auswahl Ihres Microcomputer-Systems will gut überlegt sein. Schließlich soll der Computer ja Ihr individueller Problemlöser werden.

Lassen Sie sich also nicht vom erstbesten Anbieter einwickeln!

Leisten Sie sich die Qual der Wahl und das Vergnügen, von Fachleuten beraten zu werden. Wir sind nämlich Vertragshändler aller namhaften Microcomputer-Hersteller. Bei uns sollen Sie probieren, vergleichen, testen – und müssen keine Auswahl per Postkarte treffen.

Auch mit Ihren Service-Problemen sind Sie bei uns in guten Händen. Eine komplett ausgerüstete Werkstatt mit Hersteller-geschulten Ingenieuren bringt Ihren Computer schnell wieder in Ordnung.

Und mit Zubehör und Peripherie-Geräten aus unserem Sortiment machen Sie im Handumdreh'n aus Ihrer Zentraleinheit ein Microcomputer-System.

Gibt es bessere Gründe für einen Besuch?

micro Computer-Zentrum — Die **Nr. 1** in Hessen

Darmstadt

Alsfelder Straße 7 · Am Meßplatz
6100 Darmstadt · Tel. 06151/76032

Frankfurt

Dreieichstr. 59 · Am Lokalbahnhof
6000 Frankfurt 70 · Tel. 0611/625048

geöffnet: montags – freitags von 8.30 – 13.00 und 14.00 – 18.00 Uhr, jeden 1. Samstag im Monat von 10.00 – 16.00 Uhr

---

Besuchen Sie uns Mo.–Fr. von 8–12 Uhr und von 13.30–18.00 Uhr. Sa. von 8–12 Uhr

Apple

6006	DATA-Management	78,00
6017	Inventurprogramm	299,00
6014	The Basic Teacher	84,00
6011	Invoicing	189,00
6002	Priv. Sekretärin	189,00
6015	Billing Management	299,00
6016	Retail Management	189,00
6010	Asset Record	189,00
6007	Progr. Gymnastik	63,00
6111	Mailing-List	249,00
6001	Prof. Sekretärin	299,00
6013	Word Processor	299,00
6110	Apple Sargon (C)	110,00
6118	Apple Sargon (D)	119,00
6119	Super FORTH	169,00

Neue Apple Software D=Diskette, C=Cassette

6120	Reversal (D)	129,00
6126	Dateiverwaltung (D)	199,00
6127	Adressenverwaltung (D)	199,00
6128	Super Invaders (D)	49,00
6129	PASCAL Programme (D)	29,80
6230	Utilities I (D)	99,00
6131	Utilities II (D)	99,00
6132	Statistik (D)	99,00
6133	Inventory (D)	69,00
6134	Inviocing (D)	79,00
6135	Dictionary (D)	49,00
6136	Game Package (D)	69,00

HAYDEN

254	The S-100 Handbook	49,00
255	BASIC BASIC	39,00
256	Stimulating Simulations	19,80
257	BASIC Comp. Progr. in Science and Engineering	39,00
258	APL-An Introduction	39,00
259	Creative Progr. for Fun and Profit	29,80
260	BASIC Comp. Progr. f. Business, I	39,00
261	BASIC Comp. Progr. f. Business, 2	39,00
262	Homecomputer can make you rich	19,80
263	Sixty Challang. Problems	19,80
264	The complete 1802 Cookbook	19,80
265	Musical Applications for Micros	79,00
266	Advanced BASIC Appl.	39,00
267	How to profit from your Microc.	39,00
268	Pascal with Style	39,00
269	Cobol with Style	39,00
270	BASIC with Style	39,00
271	BASIC FORTRAN	45,00
272	Z80 and 8080 Assembly Language Programming	39,00
273	Beat the ODDS: Microcomputer Simulations of Casino Games	39,00

MÜNZENLOHER GMBH

Tölzer Straße 5
D-8150 Holzkirchen/Obb.
Tel.: (0 80 24) 18 14

Gerätebau, Computersysteme – Software
Lieferung per NN oder Vorkasse auf Postscheckkonto 2845 58-807 München oder Euroscheck

ELCOMP-Erweiterungsplatinen

(Platinenservice zu den Artikeln in ELCOMP) Wir liefern die Platine mit Anleitung und wenn notwendig die zugehörige Software. ELCOMP hat für seine Leser ein neuartiges Erweiterungssystem für alle 6502 Microcomputersysteme entwickelt. Grundlage dieses Konzeptes ist die ELCOMP 1 Expansionsplatine. Sie ermöglicht es, jeden 6502-Computer so zu erweitern, daß Sie
1. 4 Steckplätze f. Platinen m. Apple Bus und
2. 1 Steckplatz für S44-Bus erhalten.

Die Philosophie dieses Erweiterungssystems liegt darin, daß alle 6502 Systembesitzer Platinen mit dem so weit verbreiteten Apple II Bus verwenden können. Das Angebot an Erweiterungsplatinen für den Apple II ist riesig und das kann jetzt jeder 6502-Systembesitzer für sich ausnutzen. Zum Einstecken in die ELCOMP 1 Expansionsplatine haben wir zur Zeit folgende Karten fertiggestellt:

ELCOMP Universal Expansion Boards

Expansions-Platinen f. Apple II, Apple II plus und Ohio Scientific Superboard C1P/C4P etc.

604	Universalexperimentierplatine	59,00
605	Ein-/Ausgabe Experimentierpl.	89,00
606	Bus Expansion ELCOMP 1	129,00
607	EPROM-Platine f. 2716-Burner	149,00
613	8K RAM-Karte (nur über Bus Expansion anschließbar	69,00
612	32K RAM-Karte dynamisch	169,00
608	Musik-Platine für GI-AZ 8912	89,00
609	EPROM-Karte für 1 2716	59,00
610	Analog-Digitalwandler-Karte	149,00
611	Rechnerkopplung Apple II/Superboard, 2 Platinen mit Anleit. u. Softw.	249,00

---

r+r electronic

Adlerstraße 55, 6900 Heidelberg 1, Tel. 0 62 21/83 31 29

Das besondere Angebot:

AIM-Benutzerhandbuch in Engl. ..... DM 7.50
bei Vorauszahlung auf PschKto. KA 805 64-753

AIM-65
4-KByte-RAM
1150.–
4-KByte-RAM
1075.–

inkl. engl. Benutzerhandbuch
Aufpreis dt. Benutzerhandbuch + DM 10.–
Benutzerhandbuch in Deutsch ..... DM 29.80
AIM-Federleiste 44polig ..... DM 9.95
Thermopapier Pack 10 Roll., à 25 m DM 29.50
AIM-Netzteil 5 V/5 A, 24 V/0,5 A .... DM 195.–
AIM-Metallgehäuse ..... DM 249.–
Software v. Siemens auf CC 1, AIM-65 u. PC 100
Spiele 1, Spiele 2 ..... je DM 55.–
Mathematik 1, 2, 3 ..... je DM 84.50
Statistik 1, Finanzmathematik ..... DM 84.50
Software für AIM-65 u. PC 100 in ROM
4-K-Assembler (ROM) ..... DM 285.–
8K-Basic (2 ROMs und Manual) .... DM 340.–
Neu, PL-65 Compiler (ROM) ..... DM 425.–

AIM-65 Komplettpaket

bestehend aus folgenden Teilen:
AIM-65 4 KB mit dt. Buch, 8KB-BASIC und Manual (engl.), Netzteil 5 A, Metallgehäuse, Federleiste. 10er-Pack Thermopapier und 10 Datenkassetten
Komplett-Paketpreis ..... DM 1875.–

PC-1211

BASIC-Taschenrechner ..... DM 420.–
Cassetten-Schnittstelle ..... DM 63.90
Drucker f. PC 1211 ca. 5/81 ..... DM 275.–
Fordern Sie weitere Informationen an:

SORCERER

Standardausführung: 16 bis 48-KB-RAM, Z-80 CPU, Groß- u. Kleinschreibung, Grafikmöglichkeit, serielle u. paraliele Schnittstelle, Tastatur mit Zehnerblock. Option: Floppy-Laufwerke bis 630 KB m. CB/ u. MBasic.
Programmierbar in COBOL, FORTRAN und PASCAL.
Unsere neueste Gesamtpreisliste erhalten Sie gerne auf Anforderung.

Daten-Displays

12"-Monitor. 560.–
Monitor grün. Bildschirmgr. 12" (31 cm) professionelle Ausführung für kommerziellen Einsatz.

12"-Monitor. 765.–
Sanyo-Datensichtgerät DM 5912 CX mit grüner Bildröhre und extrem hoher Auflösung (18 MHz). Abm. 320 × 280 × 305 mm. Der Monitor für den anspruchsvollen Anwender.

12"-Monitor. 825.–
12"-Daten-Display DM 5112 CX mit grüner Anzeige in professionellem Metall-Gehäuse mit Kunststoff-Front. Verdeckte Bedienelemente. Bandbreite 18 MHz.

8" + 5,25"-Disketten
nur Spitzenqualität BASF. 2 Ausführungen soft, sekt. oder handsekt. m. 10 bzw. 32 Sekt.
1 St. DM 9.90 **10 St. DM 73.95**

8"-Disketten 2D
BASF, Double Density 2seitig z. B. für Apple 8"-Laufwerk, **1 St. DM 23.95** **10 St. DM 199.–**

Tabellierpapier
240 mm × 12" 1 Stachelabstand 9"/229 mm. Ausführung einfach weiß oder grün/weiß, z. B. f. Centronics 730, Anadex, Adoomp oder MX 80. Verpackungseinheit je Sorte 2000 Blatt
1 Karton = 2000 Blatt ..... DM 49.–

Datenkassetten
Unsere Datenkassetten enthalten nur hochwertiges Baumaterial in einem fünffach verschraubten Gehäuse mit zweiteiliger Schutzbox. Mit großen weißen Aufklebern für Ihre persönliche Beschriftung versehen. Länge ca. 15 m (ca. 5 min/Seite).
Wir liefern nur Spitzenqualität.
Neu! Jetzt ohne Vorspannband.
10 St. = DM 18.95 **100 St. = DM 169.50**

Adreß-Etiketten Neu
Größe: 100 × 35,5 mm in 2 Ausf. lieferbar. 1bahnig 122 × 12" od. 2bahnig 227 × 12". Mindestabnahme 1 Karton mit 4000 bzw. 8000 Etiketten.
Karton (1bahnig) 4000 ..... DM 66.50
Karton (2bahnig) 8000 ..... DM 135.60

Apple II Plus Europlus

Das Computersystem für alle kaufmännischen und technischen Bereiche wie auch für den anspruchsvollen Hobbyisten.
APPLE II PLUS mit 16-KB-RAM **DM 2675.–**
APPLE II PLUS mit 32-KB-RAM **DM 2785.–**
APPLE II PLUS mit 48-KB-RAM **DM 2895.–**

Neu endlich lieferbar:

Apple II Plus mit 64 KB-RAM **DM 3335.–**
Erweiterungen für APPLE o. ITT 2020:
1. Disk-Laufwerk (mit DOS 3,3) ..... DM 1760.–
PAL-Modulator ..... DM 325.–
S/W-Modulator ..... DM 64.–
16 KB-RAM-Karte ..... DM 495.–
Controller f. 8"-Laufwerke ..... DM 1325.–
8"-Laufwerke bis 1 MB ..... DM 3545.–
Gesamtpreisliste bitte kostenlos anfordern.

System A . . . DM 3385.–
Apple II Plus 48 KB-RAM mit 12"-Monitor grün

System B . . . DM 4950.–
Apple II Plus 48 KB-RAM mit 1. Floppy Laufwerk 143 KB und 12"-Monitor grün.

System C . . . DM 6735.–
Apple II Plus 48 KB-RAM mit 1. Floppy-Laufwerk 143 KB, 12"-Monitor grün und EPSON MX 80 F/T inkl. Apple-Interface und Kabel.

Apple-Software

Textverarbeitung ..... **DM 250.–**
Lagerhaltung ..... **DM 540.–**
Kundenkartei ..... **DM 525.–**
Basic-Kurs ..... **DM 148.–**
VISICALC ..... **DM 495.–**
Cashier (Laherh./Fakt.) ..... **DM 1090.–**
Adressenkartei ..... **DM 495.–**
Statistik ..... **DM 990.–**
Apple Pilot ..... **DM 405.–**

APPLE-Neuheiten

80 × 24 Z/Z-Karte ..... **DM 789.–**
einsetzbar für BASIC und PASCAL
Z-80 Softcard ..... **DM 868.–**
inkl. CP/M 2.2 und BASIC 80
Tastaturumrüstsatz ..... **DM 245.–**
Endlich! Groß-/Kleinschreibung. Umlaute, Shift, Tastenfunktion, Alphalock und und ...
DOS 3.3 set ..... **DM 165.–**
Aufrüstsatz für DOS 3.2 inkl. Manual
DOS Tool Kit mit Manuals ..... **DM 189.–**
Assemler/Editor Hi-res Character Generator Renumber. Zeilennumerierung und viele nützliche Programme auf Diskette.
APPLE-Pilot ..... **DM 405.–**
Ein wichtiges Instrument f. alle die unterrichten u. lehren. Ideal zur Erstellung programmierter Lehrprogramme.
APPLE-Plot ..... **DM 186.–**
Damit können Sie wichtige Daten graphisch darstellen, ändern, analysieren und ausdrücken.
Fordern Sie unsere Neuheiten-Info an.

ITT 2020 PALSOFT

ITT 2020 16K m. PAL-Karte ..... DM 3199.–
ITT 2020 48K m.PAL-Karte ..... DM 3399.–
Zubehör siehe APPLE II PLUS.

Video-Genie EG 3003

Neu! Jetzt mit Cursortasten
- Z-80, 12-KB-ROM, 16-KB-RAM
- eingebauter CC-Recorder u. Netzteil
- UHF + Video-Ausgang
- 16 × 64/16 × 32 Zeichen
- über Expansions-Interf. f. Floppy erweiterb.
- Centronics Druckerschnittstelle

Komplettpreis . . . 1395.–
NEU NEU NEU

Video-Genie EG 3008

Jetzt gibt es den großen Bruder des EG 3003. Anstelle CC-Recorder, 10er-Tastatur und 4 Funktionstasten. Sonst gleiche techn. Daten.
Komplettpreis ..... DM 1590.–
Auf Wunsch ohne Aufpreis:
Nur bei uns werden alle Video-Genies mit verbessertem UHF-Modulator geliefert. Gesamtpreisliste bitte anfordern.

MX 80 F/T. 1675.–
– Normalpapier, Friktions-/Traktortyp
– Einzelblatt, Rollen- und Endlospapier
– Papierbreite von 4" bis 10" verstellbar
– sonstige Daten wie MX 80
MX 80 F/T inkl. Schnittstelle für APPLE, TRS 80, CBM, Video-Genie oder RS-232. ..... 1825.–

MX 80. . . . . 1525.–
– Normalpapier, Traktorführung verstellbar
– 40, 66, 80 u. 120 Zeichen/Zeile
– 80 Z/s bidirektionale Druckwegoptimierung
– 96 ASCII + 64 Graphik + 8 internationale Zeichen umschaltbar
– Interface für sämtliche Schnittstellen lieferbar (Option)
– 15 programmierbare Funktionen
– gutes Schriftbild durch 9 × 9-Matrix
MX 80 inkl. Schnittstelle für APPLE, TRS 80, CBM, Video-Genie oder RS-232 ..... **1725.–**

MX 80 Mod. II a. Anfr.
Ausf. wie MX 80 mit Apple-Interface und zusätzlich volle Graphik-Möglichkeit.

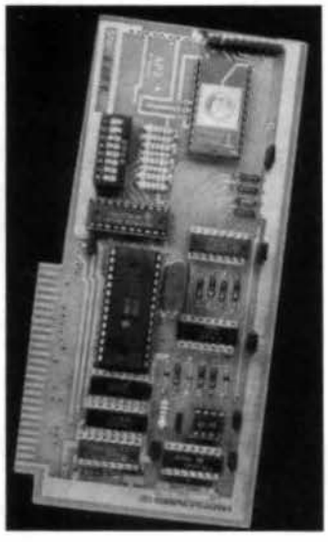
MX 80 F/T Mod. II a. Anfr.

Ladenverkauf: Adlerstraße 55, 6900 HD-Wieblingen
Geschäftsz.: Mo.-Fr. 9-13, 14-18, Sa. 9-13. Preise incl. Mwst.
Versand per Nachnahme ab DM 30.– + Versandkosten
Sendungen ins Ausland nur per Vorausrechnung
Preisänderungen und Zwischenverkauf vorbehalten.

NEU

Apple-Interface-Karten

Eine Serie von Interface-Karten für Apple-II-Anwender stellt die IBS Computertechnik vor. Durch die Verwendung von LSI-Bausteinen erhält der Apple II Leistungsmerkmale, auf deren Bedeutung Apple



bei der Vorstellung des Apple III hingewiesen hat. Schon jetzt stehen für den Apple II ein serielles Interface mit dem 6551 und ein Parallel-Interface mit dem 6522 zur Verfügung.

Zwei weitere interessante Entwicklungen sind eine 16-KByte-RAM-Karte und eine neue PAL-Farbkarte. Die 16-KByte-RAM-Karte macht aus jedem 48-KByte-Apple einen 64-KByte-Rechner und ist austauschbar mit der von Apple vertriebenen Language-Karte. Durch das neue DOS 3.3 in Verbindung mit der 16-KByte-Karte wird automatisch das schnelle Integer-Basic in den Apple-II-Plus geladen und steht parallel zu Applesoft zur Verfügung.
Die PAL-Karte ist, genau wie die bereits genannten Karten, eine Eigenentwicklung.
Zur vollständigen galvanischen Trennung für Leistungssteuerungen mit dem Apple wird eine Vier- oder Achtfach-Relaiskarte angeboten; für alle, die selbst den Lötkolben in die Hand nehmen, steht eine Experimentierkarte zur Verfügung.
(IBS Computertechnik, Postfach 14 08 69, 4800 Bielefeld 14)

IBM muß nicht teuer sein

Für bisherige Maßstäbe verhältnismäßig preisgünstig (ca. 3000 DM) ist das von IBM gebaute Bildschirm-Terminal 3102/12. Es besteht aus drei einzelnen, zusammensetzbaren Teilen, dem 12-Zoll-Bild-

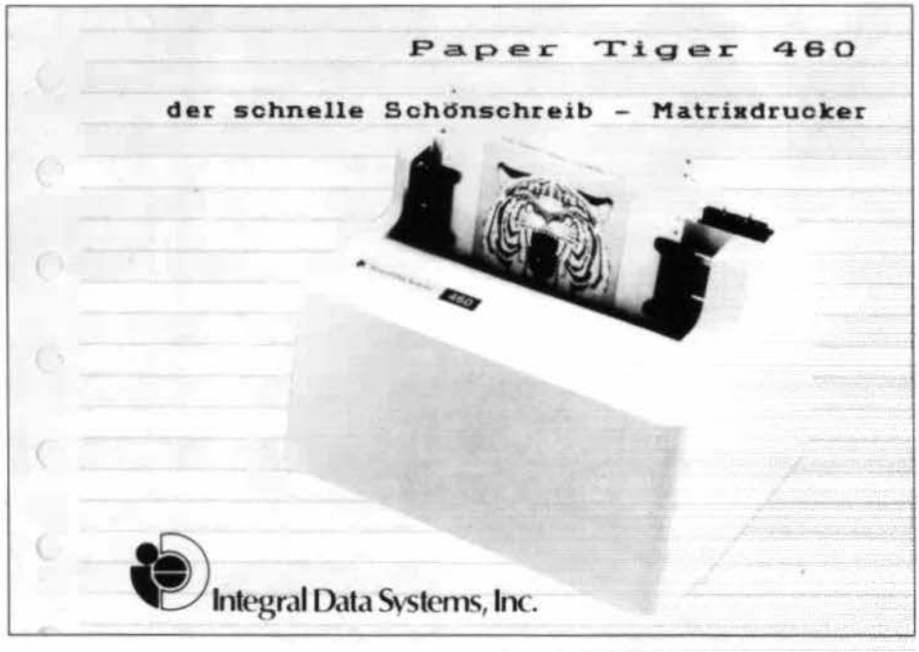
schirm, dem Bodenteil mit der Elektronik und dem absetzbaren Tastenfeld. Die wählbare Darstellung des Schriftbildes – grün auf dunklem Grund oder schwarz auf hellgrünem Grund – sowie ein spezielles abnehmbares Kontrastfilter eleminieren weitgehend die bekannten Ermüdungserscheinungen an herkömmlichen Bildschirm-Arbeitsplätzen.
Das Bildschirm-Terminal IBM3101/12 kann mit den IBM-Systemen 370, IBM303X, IBM4300, dem IBM8100-Informationssystem und der IBM-Serie/1 betrieben werden sowie über die V24-Schnittstelle oder die 20-mA-Stromschleifen-Schnittstelle mit den meisten Mikroprozessor- und Host-Rechner-Systemen zusammengeschaltet werden.
Die Datenübertragung erfolgt beim 3101/12 nach dem „Character-Transmission-Mode". Alle anderen Modelle der Typenreihe 3101 sind auf Anfrage ebenfalls verfügbar. Das Bildschirmterminal 3101 verfügt über eingebaute Eigentestfunktionen, deren Meldungen in einer zusätzlichen Zeile (Zeile 25) auf dem Bildschirm dargestellt werden. Der Eigentest wird durch die entsprechende Schalterstellung NORMAL/TEST eingeleitet. In der Stellung TEST werden interne Tests ausgeführt; der Zeichensatz von 95 Zeichen sowie die Stellung der DIP-Schalter für die Optionswahl werden dargestellt. Über das Tastenfeld können Zeichen eingegeben werden.
Mit den Optionswahlschaltern oberhalb des Tastenfeldes können u. a. folgende Funktionen eingestellt werden: Halb-/Vollduplex; V24/20-mA-Stromschleife; permanenter Request-to-Send/gesteuerter Request-to-Send; mono/dual; 1 und 2 Stop-Bits; Parität: Space/Mark/Odd/Even; Nullunterdrückung; automatische neue Zeile; automatischer Zeilenverschub; CR/LF-Kombination; Reverse Video-Darstellung; blinkender Cursor; ein-

2 Gegensätze erstmals in einem Drucker vereint

- Korrespondenzfähiges Schriftbild durch überlappende 9 x 24 Matrix
- bis 160 Zeichen/s im Dauerbetrieb durch bidirektionales Drucken mit Wegoptimierung
- 12 verschiedene Schriftbilder per Software programmierbar; wahlweise in Proportional- oder Konstantschrift
- stark reduzierter Software-Aufwand durch automatischen Randausgleich, definierbare Textbreite, horizontale und vertikale Tabullierung
- durch serienmäßig vorhandene V 24- und Centronics-Schnittstelle leicht an alle Rechner anschließbar (z.B. Apple, TRS 80, CBM u.v.a.)

ZIEGLER 41

ZIEGLER-Instruments GmbH & Co.
Bereich Meßtechnik
4050 Mönchengladbach 2, Postf. 510, Tel. 02166-80091



stellbare Baudrate (110, 150, 200, 300, 600, 1200, 1800, 2400, 4800, 9600).
Der Videoteil arbeitet mit einer Bildwechselfrequenz von 70 Hz. Es sind 1920 Zeichen (80 Zeichen/24 Zeilen) nach dem ASCII-Zeichensatz (große und kleine Buchstaben) darstellbar. Das Selectric-Tastenfeld mit 87 Tasten, 12stelligem numerischem Tastensatz und den Bedientasten für die Cursor-Steuerung ist übersichtlich und benutzerfreundlich gestaltet. Die Parität der Empfangsdaten, Tastenfeldüberlauf, Data-Set-Ready-Pegel und Clear-to-Send-Pegel werden kontinuierlich durch den eingebauten Mikrocomputer überwacht.
(Spezial Elektronik KG, Postfach 1308, 3062 Bückeberg)

Olivetti-Interface

Speziell für die elektronisch ansteuerbaren Olivetti-Typenrad-Schreibmaschinen entwikkelte MicroComp ein Interface für die Verbindung mit Mikrocomputern, das in unterschiedlichen Versionen (für die üblichen Schnittstellen 8 Bit parallel, RS-232, V-24, IEC-Bus) zu haben ist. Damit wird die Schreibmaschine zu einem

komfortablen Ein- und Ausgabegerät z. B. für die Textverarbeitung, ohne ihre normale Funktion als Schreibmaschine zu behindern. Der deutsche Zeichensatz ist Standard. Für nicht standardmäßige Typenräder können die entsprechenden Zeichensätze in das Interface einprogrammiert werden. Das Interface enthält einen Datenpuffer von 1024 Byte. Die Übertragungsraten lassen sich von 50 bis 19 200 Baud einstellen. Die Druckgeschwindigkeit beträgt ca. 30 Zeichen pro Sekunde. Die vielen Sonderfunktionen der Olivetti-Typenrad-Schreibmaschinen ET 121, 201, 221 und 231 sind vom Computer ansteuerbar.
Durch die hohen Schriftqualitäten und den günstigen Preis sind die Typenrad-Schreibmaschinen eine echte Alternative zu Matrix-Druckern bzw. den großen Hochgeschwindigkeits-Schönschreibdruckern.
(MicroComp, Schwerinstraße 6, 4000 Düsseldorf 30)

Textverarbeitung mit dem Apple II

Vom Apple-Vertrieb Basis Mikrocomputer ist für den Apple II neuerer Serie (ab Rev.-Level 7, Auslief. etwa ab Juli 1980), der unter der Tastatur über eine besondere Encoder-Platine verfügt, eine Europa-Version der Encoder-Platine nebst einem neuen Zeichengenerator-EPROM (vier Zeichensätze) erhältlich. Nun sind „richtige" Groß- und Kleinschreibung, deutsche Umlaute, Shift-Funktion, Alpha-Lock, Autorepeat mit jeder Taste, vier mit Drahtbrücken wählbare Zeichensätze und auch alle ASCII-Zeichen möglich; ferner sind die Buchstaben Y und Z an der von Schreibmaschinen her gewohnten Stelle. Der Umbau ist ohne Löt- und Verdrahtungsarbeiten möglich und wird von einer ausführlichen Einbau- und Bedienanweisung unterstützt.
(Basis Mikrocomputer, Friedrich-Ebert-Straße 137, 4400 Münster)

Grafik-Utility für CBM 3000

Vierzig neue, bildschirmorientierte Basic-Befehle ermöglicht ein EPROM namens „Pic-Chip".
Der gesamte Bildschirm oder ein vordefiniertes Sichtfenster innerhalb des Schirms kann um N Stellen in jede der vier Hauptrichtungen gerollt oder geschoben werden, wobei beim Schiebemodus C das

Ein Bild sagt mehr . . .

Hardware:
Superschneller Tischrechner mit 128k-RAM und 360/720k Diskspeicher sowie 7 Schnittstellen

Peripherie:
Schönschreib-Graphik-Printer zur Hardcopy des Bildschirmes, Plotter (A3), Digitizer (Auflösung 0,025 mm) und Lichtgriffel

Sprachen:
(unter CP/M): FORTRAN, BASIC, PASCAL, ALGOL, COBOL, dazu Textverarbeitungs- und Sortierprogramme

Graphik:
- Interaktives CAD-Paket zum Erstellen, Ändern, Archivieren von Zeichnungen, Überlagern mehrerer Bilder, Zoomen, Rotieren, Kurven-Digitalisieren, Längen-, Flächen-, Volumen-Berechnen und Akkumulieren u.v.a.
- komfortables CALCOMP®-Graphik-Paket für technisch-wissenschaftliche Anwendungen mit Funktionen wie SCALE, LINE, AXIS, SYMBOL, FACTOR, PLOT . . .
- Nützliche Graphik-Libraries wie Balkendiagramme, SW-Schrift, Kreise, Elypsen, Treiber für o.a. Peripherie etc.

ZIEGLER 43

ZIEGLER-Instruments GmbH & Co.
Abt. Computer
4050 Mönchengladbach 2, Postf. 510, Tel. 02166-80091

Zeichen darstellt, das die freiwerdende Fläche auffüllt. Alle Zeichen können in Groß- oder Kleinschreibung erscheinen und ihre Schreibweise kann umgekehrt werden. Die Zeichendarstellung kann normal, reverse oder inverted erfolgen. In doppelter Schreibdichte (50 × 80 Punkte) können Linien zwischen zwei beliebigen Punkten gezeichnet oder gelöscht werden, oder auch von einem Punkt senkrecht zu jeder der beiden Achsen. Einzelne Punkte können gezeichnet oder gelöscht werden. Alle X/Y-Koordinaten sind auf einen Ursprung in X0/Y0 bezogen, der vom Benutzer definiert wird. Eine Möglichkeit für feine Dichte ist ebenfalls vorgesehen. Sie erlaubt es, jederzeit 25 Werte mit einer Auflösung von 320 zu zeichnen.
Der gesamte Bildschirminhalt kann jederzeit in jede oder von jeder RAM-Adresse kopiert werden. Damit können schnellwechselnde Bilder produziert werden. Zeichen können auf jeden Koordinatenpunkt geschrieben werden. Die momentane Position der Schreibmarke kann direkt in die Variablen X und Y geschrieben oder dort gelesen werden. Eine nützliche Repeat-Key-Funktion kann unter Programmsteuerung an- oder ausgeschaltet werden, ebenso wie das gesamte PicChip-Repertoire.
(Ing.-Büro Houghton, Arabellastr. 13, 8000 München 81)

Intelligente Terminals

Der amerikanische Mikrocomputer-Hersteller SWTP (Southwest Technical Products) stellt neue intelligente Terminals vor. Zwei Versionen der mit einem 6802-Mikroprozessor ausgerüsteten Geräte sind lieferbar: CT-8212, ein intelligentes Terminal mit 12-Zoll-Bildschirm, und die Version CT-8209 mit 9-Zoll-Bildschirm.
Beide Terminals sind elektrisch identisch. Über 140 Funktionen sind softwaremäßig programmierbar. Das Standard-Format 24 x 80 wird voll unterstützt. Ein eingebauter Mikroprozessor übernimmt zusammen mit dem Motorola-Video-Controller Funktionen wie Insert, Delete, Scroll, Roll, Slide usw. Die Funktionen sind lokal bedienbar und fernsteuerbar. Ein Parallel-Printer-I/O-Port ist standardmäßig eingebaut.
Die Zeichendarstellung erfolgt in einem hochauflösenden 7 x 12-Raster (mit Unterlängen, Groß- und Kleinschrift) auf einem grünen Motorola-Bildschirm. Die Übertragungsrate ist im Bereich 50...38 400 Baud programmierbar.
Weitere Eigenschaften sind schreibgeschützte Fehler, Inversdarstellung, Hervorhebung durch doppelte Helligkeit, separate numerische/Cursor-Tastatur (softwaremäßig wählbar) und Pseudografik.
(Digicomp AG, Werstr. 36, CH-8004 Zürich, Schweiz)

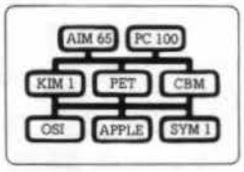
AIM-65-/PC-100-Videokarte

Verhältnismäßig preiswert ist eine Video-Karte für die nur mit einem einzeiligen alphanumerischen LED-Display ausgerüsteten Mikrocomputer AIM-65 (Rockwell) und PC-100 (Siemens). Sie liefert ein Bildschirmformat von 16 Zeilen mit je 40 Zeichen und kann direkt an einen handelsüblichen Fernsehempfänger angeschlossen werden, da sie über einen UHF-Modulator verfügt. Auf der Platine ist 1-KByte-Video-RAM untergebracht; dessen Adressenbereich läßt sich mit Schaltern beliebig festlegen. Durch Einstecken von ICs läßt sich eine Blockgrafik-Anzeige nachrüsten, die eine Auflösung von 64 × 80 Punkten auf dem Schirm ergibt, so daß sich auch Kurven hinreichend fein darstellen lassen. Der Stromverbrauch der Karte beträgt 500 mA bei 5 V. Die Anschlußbelegung entspricht der Mutterplatine aus der Microtan-65-Familie.
Die Platine enthält (z. B. im Gegensatz zu dem von Siemens angebotenen Video-Interface) keine Betriebssoftware. Es bleibt also dem Anwender selbst überlassen, ein kleines Betriebsprogramm zu schreiben, das Zeichen in das Video-RAM schreibt.
(Joachim Müller, Georg-Büchner-Str. 16, 6500 Mainz 42)

Hochauflösende Grafik bei PET und CBM

Ein für Computer der Serien CBM 3001 von Commodore ausgelegtes Grafik-Interface entwickelte Eltec. Die zusätzliche Platine läßt sich noch leicht nachträglich in das CBM-Gehäuse einbauen und wird über vier Adressen per POKE-Befehl angesprochen: Register 1 nimmt die X-Koordinate, Register 2 die Y-Koordinate auf. Das dritte Register dient der Cursor- und Punkthelligkeits-Steuerung, und Register 4 setzt alle Bildpunkte gleichzeitig dunkel (Löschbefehl).
Da das Videosignal mit dem CBM-Zeichengenerator gemischt wird, ist es auch möglich, Schrift in die Grafik einzublenden. Ein mitgeliefertes EPROM enthält nützliche Software-Routinen z. B. zum Zeichnen von Linien oder Schreiben von ganzen Flächen. Die Grafik-Platine kann auch in Versionen für abgesetzte Bildschirme sowie für 16 Graustufen oder acht Farben geliefert werden.
(Eltec, Neubrunnenstr. 10, 6500 Mainz)

Das Buch 1-6 des 65xx MICRO MAG



COMPUTING SOFTWARE HOBBY

65xx MICRO MAG

Fachzeitschrift für die Programmierung der 65xx-Mikroprozessoren in Maschinensprache/ Assembler und für Interfaceprogrammierung.

Wer beruflich oder aus persönlichem Interesse mit einem 6502-Prozessor Programmentwicklungen betreibt, erhält mit dem laufenden Bezug dieser Zeitschrift (ggfs. mit Nachlieferung früherer Hefte) eine unentbehrliche Informationsquelle, die ihm die kostengünstigeren Fortschritte ermöglicht.
Neu: Ab April 1981 auch für 6809-Systeme!

Abonnement für 6 Hefte ab lfd. Nr. DM 49.– (Ausland DM 54.–), Endpreis.
Sammelband **„Das Buch 1–6 des 65xx MICRO MAG"**, ca. 230 Seiten o. Werbung DM 26,–.

Die Hefte 7–17 sind noch nachlieferbar zu DM 7.80/St.

Herausgeber/Bezug: Dipl.-Volkswirt Roland Löhr, Hansdorfer Str. 4, D-2070 Ahrensburg, Tel. (0 41 02) 5 58 16

Buch- und Zubehörangebote:

Thermopapier für AIM 65/PC 100 in kontrastreicher Spitzenqualität. Packung mit 8 Großrollen, zus. 520 m, preiswert **DM 50,85**

Thermokopf/Printerplatte für AIM 65 und PC 100 mit Einbauanleitung. Auffrischung des Druckes **DM 23,–**

Anwenderhandbuch für AIM 65, deutsch, Original-Rockwell **DM 32,10**

6502 Software Design von L. J. Scnalon. Das beliebte Lehrbuch für die Programmierung, ca. 270 Seiten, viele verständlich aufgebaute Beispiele **DM 29,–**

Programming & Interfacing the 6502 with experiments von L. de Jong ca. 410 Seiten, viele Schaltungsvorschläge, didaktisch gegliedert, für den Anfänger besonders geeignet **DM 48,–**

Programmierseminare und Workshops für 6502 sowie Assembler- und Interfaceprogrammierung für 6502 und 6809.
Bitte Sonderprospekt anfordern. Schulungen in Firmen auf Anfrage.

INSA
INDIVIDUELLE SOFTWARE ENTWICKLUNG

Jürgen Ehrmann
Tel. (0 70 71) 3 46 73
Breslauer Straße 30
7400 Tübingen

Wir machen Ihrem Computer Beine!

Wir bieten **exakte Software-Problemlösungen** für Minis und Micros. Ebenso führen wir flexible Standard-Software z.B.:

- Superdateiverwaltung (10–20zeilig, wahlfreies Sortieren, Heraussuchen, Listen, usw.)
- Optionen (Textverarbeitung oder Kontoführung)
- Bestands-/Inventurlistenführung
- Adressenverwaltung für Versicherungen (Standardbriefe, Merkblätter, Bankeinzugsbelege)
- Finanzbuchhaltung

Wir beraten Sie auch gerne (und unabhängig von Händlerinteressen) bei der Hardware-Wahl.

Rufen Sie einfach an oder fordern Sie **unverb. Info** (2 DM Schutzgeb.) an!

HÜLSEWIG COMPUTER SYSTEME GmbH

HERSTELLUNG · VERTRIEB · SOFTWARE · SERVICE
VERTRAGSHÄNDLER VIELER MARKEN
MICROCOMPUTER
APPLE · CBM · DAI · HP · KONTRON · SUPERBRAIN SPECIAL · TANDY · VIDEO-GENIE
DRUCKER
EPSON · CENTRONICS · CBM · OLYMPIA · ITOH
LAUFWERKE
DISKETTEN-LAUFWERKE 5,25" und 8" · FESTPLATTEN 10 MByte
SPECIAL-INTERFACES
EDV-ZUBEHÖR
DISKETTEN BASF · MAXELL
KEIN GRAUIMPORT **DEALER INQUIRIES DESIRED**

Am Wünnesberg 9, 4300 Essen 1, Tel. (02 01) 71 39 04

SALHÖFER-DATENTECHNIK

Jean-Paul-Str. 19, 8650 KULMBACH

TRS-80 Level 1, 8K	1195.–	Assembler f. SYM-1 (ROM 8K)	222.50
TRS-80 Level 1,16K	1590.–	AIM-65 (1K)	998.–
TRS-80 Level 2, 4K	1490.–	AIM-65 (4K)	1099.–
TRS 80 Level 2,16K	1698.–	Basic Interpret. f. AIM-65 (ROM 8K)	255.–
TRS-80 Expansions-Interface	895.–	Assembler f. AIM-65 (ROM 4K)	310.–
Apple II (Europlus) 16K	2645.–	**Drucker:**	
Apple II (Europlus) 32K	2745.–	Epson TX-80	1495.–
Apple II (Europlus) 48K	2945.–	Centronics 730	1990.–
Diskettenlaufwerk f. Apple mit Controller (DOS 3.3 = 143K)	1540.–	Centronics 737	2365.–
ohne Controller	1190.–	**Floppy-Disk:** 5¼"-Laufwerk 100K	1120.–
UHF-Modulator f. Apple	58.–	5¼"-Laufwerk 190K	1590.–
DAI-48K	2580.–	Datensichtgerät (Monitor) 12"	459.–
SYM-1	598.–	Datensichtgerät (Monitor) 9"	365.–
Basic Interpreter f. SYM-1 (ROM 8K)	222.50	**Cassetten,** C-10, C-30, C-60, 1. Wahl!	

Wir liefern nur Qualitäts-Fabrikate!
1 St. **1.95** ab 5 St. **1.70**

Disketten, BASF oder Maxell, 1. Wahl!
5¼ Soft- oder Hardsektoriert **7.95**
8¼ Soft- oder Hardsektoriert **7.95**

Tabellierpapier
wie oben, jedoch mit 1 zusätzlichen Durchschlagblatt, zwischen 1. und 2. Blatt ist Kohlepapier eingelegt. 1 Karton = 1000 Blatt **59.50**

Adreß-Etiketten
Selbstklebend, weiß, einbahnig, Perforation des Endlos-Etikettenträgers nach jeweils 12", mit Stachelrand, für alle Drucker passend.

Größe	Kartoninhalt	Preis f. 1 Kart.
89 × 23,4 mm	6000 St.	39.50
89 × 36,1 mm	4000 St.	39.50
107 × 48,8 mm	3000 St.	45.70
125 × 48,8 mm	3000 St.	54.50

Viele weitere preisgünstige Angebote namhafter Hersteller (Software, Hardware, Interfaces, Zubehör, Fachliteratur usw.) finden Sie ausführlich beschrieben in unserer Sonderliste „DT 15", die wir Ihnen auf Anforderung gerne kostenlos zusenden.
Alle Preise enthalten 13% Mehrwertsteuer. Versand per Nachnahme

Universal-
VIDEO-EINGANG

für alle TV-Geräte!
Besonders als Computereingänge geeignet.
Mit Netz-Trennung!
Einfacher Einbau Typ **C 1**
DM 110.74 inkl. MwSt.

EGTS-Equipment Gesellschaft für Internationale Elektronik Systeme mbH
Haingasse 14, 6000 Frankfurt 60, Tel. (0 61 94) 28 21

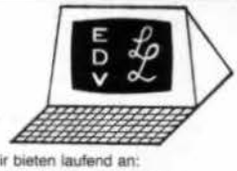
TANDY INFO

jetzt billiger als Imitationen! (it 2130) ☎ 07161-32265

CPU-»TRS-80« »Z-80« mit Volltastatur + Rechnertastatur + Netzteil + Basic »Level 2« in 14 K ROM + deutsches Handbuch + TV-Modulator + freie 16 k RAM = **DM 1.490,–**

dto. + mit Bildschirm + Recorder = **DM 1.678,–**

INFO TELECOM
INFO GMBH
7336 UHINGEN
Holzhäuser Straße 3



EDV-Lehrinstitut-LEY

Unterster Weg 61, 5024 Pulheim
Telefon (0 22 38) 5 83 30

Wir bieten laufend an:

- **BASIC-Grundkurse;** Bedienung und Programmierung ohne Peripherie
- **Drucker und Floppy** Programmierung, Dateiorganisation
- **BASIC-Aufbaukurse;** Befehlserweiterung; strukturierte Programme
- **Assembler und Maschinensprache**
- **Spezialkurse zur Entwicklungfirmenspezifischer Software**

- **nicht Demonstration, sondern aktive Teilnahme** Für jeden Teilnehmer steht eine vollständige Anlage mit Bildschirmeinheit, Drucker und Floppy zur Verfügung. Daher optimale Lerneffizienz durch ständige Fehlerkontrolle.
- **keine Massenveranstaltungen, sondern kleine Kurse** Max. 10 Teilnehmer, daher individuelle Betreuung gewährleistet. Spezielle Teilnehmerfragen können während des Kurses geklärt werden.
- **(fast) freie Terminwahl** Wochenendseminare, Abendkurse, Tageskurse, Sonderkurse sind kombinierbar.
- **Vernünftige Gebühren** Die Unterrichtsgebühren sind Endpreise inkl. Kosten für Arbeitsunterlagen, Kassetten, Disketten, Druckerpapier usw., nötige Erfrischungen während der sicherlich anstrengenden Kursdauer.
- **systemunabhängig** Der Lerninhalt ist auf nahezu jedes Computersystem übertragbar. Sie können nicht kaufen, wohl aber neutral beraten werden und vergleichen.

Aachen

GWK
GESELLSCHAFT FÜR TECHNISCHE ELEKTRONIK mbH
HARDWARE SOFTWARE SYSTEMENTWICKLUNG
FÜR 6502 UND 6809 COMPUTER SYSTEM
D 5120 Herzogenrath Astermstr. 2
Tel.: 02406 / 62394 Telex: 8321109 gwk d

RMI Nachrichtentechnik Gesellschaft mit beschränkter Haftung
TRS-80-Cromemco-EXO

1. Aachener Computerladen
Rosstr. 7, Tel. 02 41/2 40 70, Telex 8 32 521 rmi d

Aschaffenburg

CSG Das **Computerzentrum** im Untermaingebiet
Computer Service GmbH
Frohsinnstr. 32, 8750 Aschaffenburg, Tel. (0 60 21) 1 42 12

Bad Homburg

JOHANN F. BEURER
special software
Software für HEWLETT-PACKARD 9845/9835,
Meßdatenerfassung, Auswertung über Verbundnetz.
Steuerung per MIKRO.
SPRACHEINGABE am Computer.
BASIC-SEMINAR beim Anwender.
Victor-Achard-Straße 11, Telefon (0 61 72) 3 27 62
6380 BAD HOMBURG 5--Dornholzhausen

Bad Honnef

ABC80
Der professionelle Klein-Computer
Software – Zubehör
industrie design
5340 Bad Honnef 6
Himberger Straße 5a
Telefon (0 22 24) 8 01 26

Bad Kissingen

Tandy Radio Shack Vertragshändler Computer-Systeme Tel.: (09 71) 51 59
SOFTWARE - HARDWARE - UMRÜSTUNGEN - BERATUNG - SERVICE - EILVERSAND
8730 Bad Kissingen · Hemmerichstr. 10

Berlin

Rudow Büroelektronik
Keithstr. 26
D-1000 Berlin 30
☎ (030) 26 111 26
Einziger Berliner HEWLETT PACKARD Commodore Vertragshändler
Hardware · Software
Beratung · Service
Fachbücher · Seminare

Berlin

COMPUTER ELEKTRONIK
Computer und Peripherie
Sharp, Commodore
Exidy, Panasonic,
Tally-Mannesmann.
Beratung und Verkauf:
Berkaer Str. 39
1000 Berlin 33
030 - 826 16 10

Bielefeld

IBS COMPUTERTECHNIK
Artur-Ladebeck-Str. 143
4800 Bielefeld 14
Telefon 05 21/15 28 07
Computersysteme
Hardwareentwicklung
Softwareentwicklung

Bonn

Mikrocomputer von COMMODORE,
DAI, EXIDY u. a.
Peripherie, µC-Bauteile, Software,
Fachbücher und -zeitschriften
COMPUTER-SHOP BONN
Kaiserstr. 16 + 20, 5300 Bonn 1

Braunschweig

COMPUTERSTUDIO
Rebenring 50, Tel. (05 31) 34 17 34
Autorisierter Mikrocomputer-Vertragshandel
Computer v. TANDY, SHARP, APPLE, COMMODORE u. a.
BRAUNSCHWEIG

Bremen

Bremens 1. Mikrocomputerfachgeschäft
HANS SCHRÖDER
Computer-Systeme –
Systeme – Zubehör – Programme –
Zeitschriften – Bücher
Tel. (04 21) 45 97 79, Föhrenstr. 19, 28 Bremen 1

WEBER Funk
Funk · Elektronik · Computer · Video
Emil-von-Behring-Straße 6
Telefon (04 21) 49 00 10/19

Darmstadt

micro Computer-Zentrum
Die **Nr. 1** in Hessen
Alsfelder Str. 7 · 6100 Darmstadt
am Meßplatz · Telefon 0 61 51/7 60 32

Dortmund

city-elektronik
Bauteile- Funk- und Meßgeräte
APPLE, ITT-2020, CBM, SHARP, EG-3003
4600 DORTMUND 1
Güntherstr. 75 + Weißenburger Str. 43
Telefon 02 31 / 57 22 84

Düren

RÄBIGER MICROCOMPUTER, Postfach,
5160 Düren, Tel. 0 24 21/4 38 77
HARDWARE: EACA, EXIDY, TI, EPSON,
CENTRONICS, ATARI
SOFTWARE f. ATARI, EACA, EXIDY

Düsseldorf

DER COMPUTERLADEN
MIT DEM KOMPLETTEN
EDV-ANGEBOT
DATA BECKER GMBH
im Hause Auto Becker
Merowingerstr. 30 · 4000 Düsseldorf
Tel. 0211/312085/86 · Telex 08582874

Essen

HÜLSEWIG COMPUTER SYSTEME GmbH
HERSTELLUNG · VERTRIEB · SOFTWARE · SERVICE
Vertragshändler vieler Marken
APPLE · CBM · DAI · HP · KONTRON · SUPERBRAIN SPECIAL
TANDY · VIDEO-GENIE · EPSON · CENTRONICS · ITOH · OLYMPIA
SPECIAL DISK-LAUFWERKE + FESTPLATTEN · EDV-ZUBEHÖR
DEALER INQUIRIES DESIRED
Am Wünnesberg 9, 4300 Essen 1, Tel. (0201) 713904

Frankfurt

Preisvorteil + Know-how =
ComputerLand®
Ihr Spezialist für Mikrocomputer
Im Rhein-Main-Gebiet.
Buchrainstr. 34, 6000 Frankfurt/M. 70
Telefon (06 11) 65 43 33, 65 43 43

Henneve KG Moderne Büro-Organisation
3x im Rhein-Main-Gebiet Frankfurt Mainz Wiesbaden
Der autorisierte Vertragshändler von
HEWLETT-PACKARD u. COMMODORE
Zeil 127 (Hauptwache) · 6000 F.fm. · ☎ (0611) 28 14 82

Frankfurt

micro Computer-Zentrum
Die **Nr. 1** in Hessen
Dreieichstr. 59 · am Lokalbahnhof
6000 Frankfurt 70 · Tel. 0611/625048

Gießen

CENTRONICS – Vertragshändler – SHARP
electronic circuits ec (GmbH)
Gesellschaft m. b. H.
MICROCOMPUTER Verkauf – Beratung Service – Entwicklung Gebr. Computer-Peripherie
6301 Wettenberg-Wißmar · Bahnhofstr. 19 · Tel. (0 64 06) 40 63

Göttingen

Computerdienst
Brockhaus & Müller KG
Wir vertreiben
apple commodore
Weender Landstr. 3 · 3400 Göttingen
(IDUNA-ZENTRUM) · Tel. (05 51) 5 52 82

Hamburg

alphatronic
Friedrich Karl Schroeder GmbH & Co
Steilshooper Straße 293 · 2000 Hamburg 60
Telefon 040 - 63 86-1

COMPACT
Mikrocomputer GmbH
Landwehr 27
2000 Hamburg 76
Tel. 040/25 80 69
COMPACT

Mitglied des bef
Mikrocomputer + Zubehör
Speichermedien
ALFRED GRAUMANN
Elektronik-Vertrieb
Grevenweg 83, 2000 Hamburg 26
Tel. (0 40) 2 50 30 50, FS 211 768 agev d
Mitglied im Bundesverband des Elektronik-Fachhandels

HEWLETT PACKARD HEWLETT PACKARD
HP-85 Kompakt-Computer
RICHARD KIESSLING ING.
Kl. Reichenstr. 1
2000 Hamburg 11
Ruf 040/33 15 51

RIC-T RAMELOW
MRG
Informations- und Communications-Technik GmbH
Biernatzkistraße 28, 2000 Hamburg 50
Tel. 040/3 80 93 54
Hardware · Software · Entwicklung und Vertrieb
Mehr als 10 Jahre Erfahrung in der EDV-Beratung
Hardware-Interessenten fordern bitte Produktinformation an!
Schwerpunkt **Z 80 (A)-Systeme** (mit Betriebssystem CP/M + Oasis)

Hamburg

Rudow
Büroelektronik
Bachstraße 104
2000 Hamburg 76
☎ (040) 220 60 45
Commodore Vertragshändler
Hardware Software
Beratung Service
Fachbücher

Hannover

MARENO
Datensysteme
3000 Hannover – Georg-Str. 20
Tel. (05 11) 1 27 18

TCV STROETMANN **COMPUTERZENTRUM**
COMPUTER FÜR SMALL BUSINESS, HOBBY UND SCHULE
SHARP MZ 80K U. A.
SOFTWARE IN REICHER AUSWAHL/ENTWICKLUNGSABTEILUNG IM EIGENEN HAUS
3000 HANNOVER 1, PODBIELSKISTR. 129,
TEL. (05 11) 6 96 63 29
SONDERANGEBOTE + GÜNSTIGE PREISE + SONDERANGEBOTE
JEDEN MITTWOCHNACHMITTAG SPEZIAL-DEMO

Heilbronn

Computer-Markt-Heilbronn-Computer-Markt lucky dates
Wir setzen Maßstäbe in der Beratung
Am Wollhaus 1, 7100 Heilbronn,
Tel. (0 71 31) 6 93 33, Telex 7 28 273
Computer-Markt-Heilbronn-Computer-Markt-Heilbronn

Herrenberg

WILFRID HARTNAGEL
ELEKTRONIK – RECHNER
Herstellung und Alleinvertrieb von
MUSS Multi-User-Super-System für COMMODORE-Computer,
Postfach 1120, D-7033 Herrenberg 5,
Tel. 0 70 32/3 23 79, Telex 7 265 876 htsl d

Hirschau

Hauptverwaltung und Versand
CONRAD ELECTRONIC
Europas großer Electronic-Spezialist
8452 Hirschau ● Tel. (0 96 22) 19-0
Telex 6 31 205 ● Filialen:
1000 Berlin 30, Kurfürstenstr. 145, Tel. (0 30) 2 61 70 59
8000 München 2, Schillerstraße 23a, Tel. (0 89) 59 21 28
8500 Nürnberg, Leonhardstraße 3, Tel. (09 11) 26 32 80
Mitglied im Bundesverband des Elektronik-Fachhandels

Kassel

Computerdienst
Brockhaus & Müller KG
Das apple-Service-Zentrum für Südniedersachsen
Triftstraße 27 · 3502 Kassel-Vellmar,
Tel. (05 61) 82 64 47

Kempten

MIKROCOMPUTER IM ALLGÄU
Computersysteme für Beruf und Hobby
Schnelldrucker · Zubehör · Software ·
MIKROCOMPUTER-BAUSÄTZE
Ständig vorführbereit: TA-alphatronic
SHARP MZ 80, ITT 2020, EPSON MX 80
GRAF ELEKTRONIK SYSTEME GMBH
alphatronic
Magnusstraße 13,
8960 Kempten
Telefon 08 31/6 19 30
ges

Koblenz

bhg... Computersysteme
commodore – Hardware – Peripherie
apple II – Beratung – Service
Compucorp – Software – Systemanalyse
– Entwicklungssysteme
Moselring 1 · 5400 Koblenz · Tel. (02 61) 1 24 88

Köln

Colonia Computer
apple II
und andere Systeme
Centronics
Trendcom
Axiom
PASCAL, FORTRAN, PILOT, BASIC
Kundenspezifische Software Entwicklung
Colonia Computer GmbH
Lindenstr. 73/77 · 5000 Köln 1
☎ 02 21/21 57 36 · Tx. 8 885 365 ruco

Besuchen Sie
Computerland®
Ein Blick in die Welt der Kleincomputersysteme von ALTOS, APPLE, ITT 2020, North Star, Dynabyte, Cromemco mit aller Peripherie und Programmen für kommerzielle Nutzung und sinnvolle Freizeitbeschäftigung.
Eigenes Service-Center.
Computerland Köln
Blaubach 34, 5000 Köln 1
Tel. (02 21) 23 06 18

Köln

Fachgeschäft für:
antennen, funkgeräte, bauteile, computer und zubehör
IN KÖLN UND BONN
elektronik
5 Köln 80, Buchheimerstr. 23
5 Köln 1, Aachenerstr. 27
53 Bonn 1, Sternstr. 102

Krefeld

SVI
Computersysteme
Commodore, Hewlett Packard, Sharp, apple®
Peripherie
Farb-Terminals, SW-Terminals, Matrixdrucker, Typenraddrucker, Plotter, Meßgeräte
4150 Krefeld, Breuershofstr. 40
Tel. (0 21 51) 3 60 56
Software-Vertrieb-Informationsverarbeitung

Landau

DATA-SERVICE GmbH
Computer-Systeme · Software-Entwicklug · EDV-Service
- Sorcerer · Apple · CBM · System 10
- Matrix- und Typenrad-Drucker
- Individuelle Software-Entwicklung
- Komplett-Lösungen

6740 Landau/Pfalz, Kramstraße 23
Telefon (0 63 41) 8 45 77 und 67 29

Lohne

HEWLETT PACKARD
Münch Datensysteme
Brinkstraße 43
2842 Lohne
Telefon
0 44 42/25 16

München

Die Welt der Mikrocomputer
Wir führen u.a.: Apple, Altos, CBM, Diablo, TI, Cromemco, North Star mit den entsprechenden Peripheriegeräten und Programmen.
Wir bieten: Beratung und Service.
ComputerLand®
Brienner Straße 44 · 8000 München 2
Telefon: (089) 52 62 05

HOPP SPEZIAL ELEKTRONIK GMBH
- MC-Baugruppen
- MC-Software
- MC-Systeme
- automatische Meßtechnik

Ludwig-Thoma-Str. 12, 8037 Olching, Tel. 0 81 42/1 59 56

Nürnberg

apple computer
bei: **SOLARIM**
Solartechnische Gesellschaft m. b. H.
Roritzer Str. 28, Tel. 0911 / 33 48 35
8500 NÜRNBERG 90, Postf. 910 349
apple Vertragshändler Nordbayern

Paderborn

bcb computerbase
personal
commercial computersysteme
industrial
computer · import/export · verkauf · beratung · entwicklung · produktion · computer
bcb · bussmann computerbase 4790 paderborn
hathumarstr. 16, tel. (0 52 51) 2 31 35

Rastatt

SYNTAX
Soft- und Hardware GmbH
7550 Rastatt, P. B. 1609
Programme auf Kassetten und Disketten für PET und CBM

Regensburg

Jodlbauer-Elektronik
Wöhrdstraße 7, 8400 Regensburg
Tel. (09 41) 5 79 24
Computer (Hardw. + Softw.) u. Peripherie
ITT – APPLE – SHARP – DELPHIN – EPSON

Stuttgart

Arlt
Bauelemente für die Elektronik
Industrie- u. Laborbedarf
Arlt Elektronik, 7000 Stuttgart 1
Katharinenstraße 22, Fernruf 07 11 / 24 57 46

MICRO-COMPUTER-STUDIO
Ihr Partner in
STUTTGART
für das gesamte
Programm von
APPLE · TANDY
SHARP
Informations-Prospekt gratis
eschenbrenner + sauer gmbh
Ludwigstr. 87 A · Stgt.-W. · Tel.: 0711/61 22 52

Würzburg

Vector
North Star
Dynabyte
micro BAW
BAW computer systeme gmbh
Bei der Kapelle 8701 Rittershausen Tel. 09337/1044

Weilheim

Speed-up Computer Technologie GmbH
Postfach 227, 8120 Weilheim,
Telefon (0 88 02) 83 63
Cameo-Hard-Disk-Kontroller mit Laufwerk für Apple, TRS-80, Cromemco, North Star, Imsai, Heath H 89 und für die meisten S-100.

Zeitlarn

An-/Verkauf von Magnetplatten u. Datenmodulen
BASF-Disketten
RIWA-EDV + Zubehör
Hopfenweg 13 · 8411 Zeitlarn · Tel.: 09 41/6 43 52
second-hand-computer

Österreich

Innsbruck

ELECTRONIC KÖHLE GMBH
MICROCOMPUTER
Amraserstraße 1 · Telefon 3 33 61
A-6020 Innsbruck
APPLE, CBM, TRS-80, AIM 65

Wien

computer city
Wir sind die Spezialisten
Das Fachgeschäft in Sachen Hobby- und Personal-Computer.
z. B. Mikro-Computer (Apple II, CBM, TRS 80, AIM 65), Schach-Computer (Mephisto), Dolmetsch-Computer (Texas Instruments), Fachliteratur
Computer City, Hilton-Center
Landstraßer Hauptstraße 2,
1030 Wien, Telefon 75 53 82

Schweiz

Bern

COMPUTERLAND AG
Erster Computer-Shop am Platze.
Minicomputer APPLE, ITT, PET, ABC 80 Drucker CBM, EPSON, FACIT, DIABLO, OLYMPIA Plotter WATANABE, Graphic-Systems, Color-MONITOREN, VIDEO 100 usw.
Länggasstraße 28 (Nähe Bahnhof), CH-3012 Bern,
Telefon 0 31/24 25 54

Zürich

Microcomputer – Perepherien – Software – Fachbücher
microspot
Microspot AG, Sihlfeldstrasse 127
CH-8004 Zürich, Tel. 01/2 41 20 30
Montags geschlossen

BASF-Disketten-Angebot
Minidisketten 5,25", einseitig, einfache Dichte,
10 Stück DM 77.– 100 Stück DM 720.–

BASF
8", einseitig, einfache Dichte,
10 Stück DM 80.– 100 Stück DM 750.–

ODP-Minidisketten
5,25", einseitig, doppelte Dichte,
10 Stück DM 80.– 100 Stück DM 750.–
EDV-Zubehör, Ablegemappen, Disketten-Organisation usw.

Sharp MZ-80 K DM 1998.–
Sharp-Drucker P3 ..... DM 1798.–
Centronics 730-2 ..... DM 1998.–

Fortran für MZ-80,

Pascal-Interpreter für MZ-80,

Sargon-Schach 1.0
(Maschinensprache im Monitor),

Extended SP-5025,
Interpreter m. 27 neuen Befehlen,
Extended SP-6015, Interpreter,
FIBU, Lohn und Frakturierung für MZ-80.
Alle Preise inkl. ges. MwSt.

Versand gegen Scheck oder NN plus Versandkosten.

Achtung! Händlersonderliste anfordern.

Rauch-Datenverarbeitung,
Erlenstr. 30, Pf. 1842, 7460 Balingen-Erzingen, Tel. (0 74 33) 59 95,
ab 1. 5. 81 neu: 49 95, Telex 7 63 514

SHARP PC1211	DM 420.–
Drucker f. PC 1211	DM 275.–
AIM-65 (1K-V)	DM 1075.–
AIM-65 (4K-V)	DM 1150.–
Aufpreis f. dt. Handbuch	DM 10.–
8K-BASIC-Interpreter	DM 340.–
4K-ASSEMBLER	DM 285.–
Kunststoffgehäuse	DM 165.–
CHALLENGER-SUPERBOARD II:	
4K-Video	DM 745.–
4K-UHF	DM 820.–
8K-Video	DM 854.–
8K-UHF	DM 949.–
4K-RAM-Chips	DM 79.–
Netzteilbaustein	DM 84.–
Typ 610-Erweiterung	DM 889.–
CHALLENGER C4P	DM 1795.–
VIDEO GENIE (16K-V)	DM 1395.–
TCS – 32K Exp. Interf.	DM 1645.–
APPLE II + (16K-V)	DM 2675.–
APPLE II + (48K-V)	DM 2895.–
APPLE II + (64K-V)	DM 3335.–
16K-Speicherkate	DM 495.–
80-Zeichen-Karte	DM 789.–
PAL-Coder IBS	DM 308.–
DISK-II + Contr.	DM 1760.–
2. Laufwerk	DM 1240.–
Pascal + Handbücher	DM 1249.–
Z-80-Softkarte	DM 868.–
Graphic-Tablet	DM 1990.–
CASHIER	DM 1090.–
FIPAS	DM 3395.–
VISICALC	DM 495.–
ITT 2020 (16K-V)	DM 3199.–
ITT 2020 (32K-V)	DM 3299.–
ITT 2020 (48K-V)	DM 3399.–
ITT 2020 (64K-V)	DM 3849.–
SHARP MZ80K (24K-V)	DM 2045.–
SHARP MZ80K (48K-V)	DM 2449.–
SHARP ASSEMBLER-Kass.	DM 142.50
SHARP MASCHINENSPR. Kass.	DM 57.50
SHARP APPEND+RENUMBER-Kass.	DM 45.–
Commodore ab Lager	a. A.
SORCERER (16K-V)	DM 2795.–
SORCERER (48K-V)	DM 2995.–
EPSON-Drucker	
MX-80-T	DM 1525.–
MX-80-T + Interf.	DM 1725.–
MX-80-F/T	DM 1675.–
MX-80-F/T + Interf.	DM 1875.–
12" Video-Monitor grün	DM 560.–
12" SANYO-Monitor grün	DM 765.–
5¼" + 8" BASF-Disketten	DM 9.90
w. o. 10er Pack.	DM 73.95
2000 Bl. Papier 240 × 12"	DM 49.–
4000 Adress. Etikett. Aufkl.	DM 66.50

Preise inkl. MwSt. Preisänderungen und Zwischenverkauf vorbehalten. Bei Vorkasse frei Haus!

g electronic gmbh
Telex: 07 72 642 aaa-d
Habsburger Straße 134
7800 FREIBURG, Tel. (07 61) 27 68 64
Bauelemente – Bausätze – µP's
Meßgeräte – Zubehör – Fachliteratur
Fachgeschäft für Elektronik + Mikrocomputer

Floppy- + Mini-Disketten-Ordnungsmittel

Ein Diskettenkasten aus hochwertigem Kunststoff, der den Datenschutzvorschriften entsprechend verschließbar ist.

Farben: Unterteil perlweiß, Deckel anthrazit transparent.

Best.-Nr.
001.00, kl. Kasten f. ca. 30 Disk, 5,25 Zoll **DM 55.–**
002.00, gr. Kasten f. ca. 80 Disk, 5,25 Zoll **DM 76.–**
003.00, kl. Kasten f. ca. 30 Disk, 8 Zoll **DM 82.–**
004.00, gr. Kasten f. ca. 80 Disk, 8 Zoll **DM 110.–**

DISKETTEN: nur erste Wahl
10 St. in Schutztasche und Karton (in Plastikbox plus DM 9.–)

5,25",	hard oder soft,	SS SD...	**DM 85.–**
5,25",	soft,	SS DD...	**DM 94.–**
5,25",	soft,	DS DD...	**DM 135.–**
8",	soft,	SS DD...	**DM 95.–**
8",	soft (TRS80 MOD. II),	SS DD...	**DM 95.–**
8",	soft,	DS DD...	**DM 140.–**

(ohne Karton und Schutztasche minus DM 9.–)

SOFTWARE für CP/M-Betriebssysteme.
Sämtliche Software von Intersoft.
FIBU-80 für Superbrain ..... **DM 2938.–**
Wärmebedarfsrechnung in Vorbereitung. Programmier-Kurse für Basic 80, Makro 80 auf Anfrage.

Alle Preise inkl. MwSt. Preisänderung vorbehalten. Versand gegen Vorauskasse oder per Nachnahme. Bei Vorauskasse Porto und Verpackung frei.

Klaus Th. Fritz
Ing. (grad.)
Computer – Hardware – Software
Postfach 6403 D-7800 Freiburg
Sparkasse Freiburg
(BLZ 680 501 01) 1 323 535
Postscheck Karlsruhe
(BLZ 660 100 75) 858 62-756
Tel. ab 18.00 Uhr 07 61/70 04 04

PET/CBM Besitzer

Kennen Sie **SYNTAX** – das Programm-Magazin auf Kassette?

Es bringt jeden Monat 5 neue Programme in deutscher Sprache aus allen Bereichen. Zum Beispiel Datei-Systeme, Textverarbeitung, Lehrgang Maschinen-Sprache, User-Programme etc.

Kenner der **SYNTAX-MAGAZINE** loben Leistung und Preis.

Fordern Sie gleich heute noch kostenlose Informationen von

SYNTAX
Soft- u. Hardware GmbH
P. B. 1609, 7550 Rastatt
Tel. 0 72 22/3 42 96

mc

Die Mikrocomputer-Zeitschrift
bringt Programme im Strichcode

Heft 2 erscheint am 29. 6. 81.

Anzeigenschluß für Heft 2 ist der 29. 5. 81

Alles für Commodore Computer

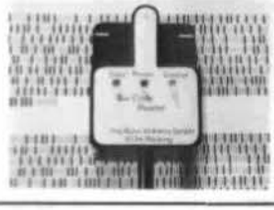
1) **BASIC COMPILER**
2) **UPGRADE SYSTEM**
erweitert CBM um 64 KBYTE auf 96 KBYTE RAM.
3) **HOCHAUFLÖSENDE GRAPHIK**
mit 64 000 frei adressierbaren Bildpunkten
4) **MULTI CLUSTER**
Computernetzwerk f. CBM bis 8 CBM Computer greifen auf eine gemeinsame Floppy zu.
5) **OLIVETTI SCHREIBMASCHINE**
anschlußfertig f. CBM (RS 232, Centronics etc. ebenfalls erhältlich)

spima computer gmbh
Turbinenstr.4 · 6800 Mannheim 31 · Tel. 06 21/72 15 15 · Telex 04 63 708 spima d

SHARP PC-1211	DM **428.–**
Personal-Computer **MZ 80K, 48K**	DM **2150.–**
sowie gesamte Peripherie auf Anfr.	
Der SUPERDRUCKER **MX 80 F/T**	DM **1777.–**
EPSON **MX 80**	DM **1626.–**
Interface zu TRS 80, Video-Genie, Apple	+ DM 150.–
SUPERBOARD II Serie 2	DM **748.–**
64 Zeichen/Zeile, Umbausatz	DM **98.–**
SORCERER Deutscher Zeichensatz	DM **198.–**

MICROPOINT
Electronic GmbH, Preise inkl. MwSt.
Alt-Griesheim 27
6230 Frankfurt 80
Tel. 06 11/38 47 42

16-K-RAM	DM **79.–**
Speichererweiterung Apple, CBM, SHARP, TRS 80 u.a.	
DISKETTEN 1. Wahl, VERBATIM	DM **79.–**
EPROM 2716	DM **30.–**
1 Spannungsvers., 3 Spannungsvers.	
APPLE Centronics Interface Board	DM **198.–**
VIDEO-GENIE-System	DM **1395.–**
TEXT-VERARBEITUNGS-**SYSTEM**	DM **6500.–**
bildschirmorientiert, komplett	



Preisgünstiger Strichcodeleser BCR1 zum Einlesen von Strichcodeprogrammen

Kein langwieriges Abtippen von Programmen mehr! Mit dem Strichcodeleser BCR1 können Sie die in der Zeitschrift mc abgedruckten Strichcodeprogramme leicht und schnell in Ihren Computer einlesen.

TTL-Ausgang, anschließbar an alle Computer, einfache 5-V-Stromversorgung, Auflösung 0,3 mm.
Preis: DM 145.– inkl. MwSt. und Porto. Lieferung gegen Vorauszahlung per Scheck oder Nachnahme.

Ing.-Büro W. Kanis GmbH, Lindenberg 113, 8134 Pöcking, Telefon (0 81 57) 16 80

mc quickie – die schnelle Produktanzeige!

mc-quickies sind aktuelle Produktanzeigen, mit denen Firmen ihre Produkte vorstellen. Verantwortlich für den Inhalt sind die Inserenten.

GWK GESELLSCHAFT FÜR TECHNISCHE ELEKTRONIK mbH

Systemexpansion für AIM 65/PC 100

- Floppy-Controller
- Video-Interface
- AD-DA-Converter
- Seriell- und Parallel-I/O
- Speichererweiterung RAM/EPROM
- EPROM-Programmer
- Prototyp-Boards
- Powersupplies-Gehäuse

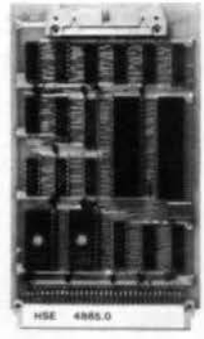
Bitte fordern Sie ausführliche Unterlagen an.

Asternstr. 2, D-5120 Herzogenrath, Telefon (0 24 06) 6 23 94

Neu: Superkram für CBM und Apple

Fängt da an, wo das File Handling Ihres Rechners leider aufhört.
Direkter Begriff auf jeden Satz, auch über mehrere Schlüssel.
Zugriff unter 1 Sekunde.
Alle Befehle in BASIC.
Dynamische Diskettenverwaltung mit physischem ERASE.
Sofort lieferbar für alle CBM-Rechner und Apple II für nur DM 498.– inkl. MwSt.

DATA BECKER GMBH
Merowingerstr. 30 · 4000 Düsseldorf
Tel. 02 11/31 20 85/86 · Telex 08 582 874



- HOPP 10.0 Listener Talker
- HOPP 4885.0 Listener Talker mit µC
- HOPP 488C.0 Listener Talker Controller
- Softwaretreiber f. Initialisierung + Betrieb

HOPP SPEZIAL ELEKTRONIK
Ludwig-Thoma-Straße 12, 8037 Olching,
Telefon (0 81 42) 1 59 56

Mikro-Computer LSI 11/2

Doppel-Floppy 9.950:DM

- Die neue LSI 11/2 ist kleiner, flexibler und schneller
- Bis 64 Kbyte ausbaufähig
- RAM–ROM–EPROM Speicher
- Mehr als 400 Instruktionen
- Gleitkomma Arithmetik
- **Preisgünstig – kurzfristig**

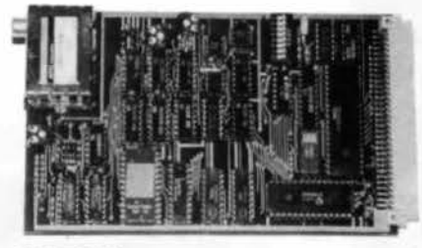
DTM DATEN-TECHNIK MÜNCHEN
DTM Datentechnik GmbH
Habsburger Platz 2,
8000 München 40
Telefon (089) 34 7011

mc quickie
Produktanzeige

Format 60 × 75 mm, Foto 35 × 55 mm
10 Zeilen Text à 40 Anschläge
+ 2 Zeilen Anschrift incl. Filmkosten DM 400.–
(ab 3 × DM 380.–, 7 × DM 360.–, 13 × DM 340,–,
26 × und mehr DM 320.–)

Anzeigen-schlußtermine

Nr. 2 vom Juli,	29. 6.	29. 5.
Nr. 3 vom Sept.,	31. 8.	31. 7.
Nr. 4 vom Nov.,	26. 10.	25. 9.



VIDEO III

Universelle Videointerface auf Eurokarte
Progr.-Zeichensatz, z. B. ASCII groß und klein
Serielle Schnittstelle 50–19200 Bd.
Parallele Schnittstelle für Tastatur
Zwei einstellbare Helligkeitsstufen
Umschaltbare Schriftbreite, invertiertes Bild u. Blinken, alles Softwaresteuerung
Optionen: Erw. Zeichensatz, UHF-Modulator
Sensationeller Preis: DM 398 inkl. MwSt.
Ostrowski Mikroelektronik,
Zum Römergrund 59, 6501 Wörrstadt,
Telefon (0 67 32) 41 78, nachm.

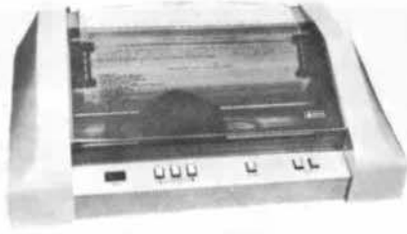
NEU: ANADEX DP-9000/9001

Die „Kompakten" mit der großen Leistung:

- 80/100/120/132 Schreibstellen
- Papierbreite 44–240 mm stufenlos

sonst wie DP-9500/9501
Bitte fordern Sie Unterlagen an.

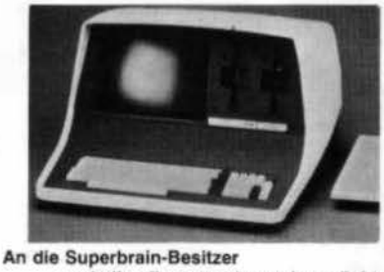
NEUMÜLLER MESSTECHNIK
Eschenstraße 2 8021 Taufkirchen/München
Telefon 089/61 18-1 Telex 5-22106



ANADEX DP-9500/9501

- 132/165/198/220 Schreibstellen
- 7x9, 9x9 oder 11x9 Matrizen
- Druckgeschwindigkeit bis 200 Zeichen/s
- Graphik-Auflösung bis 75 dots/inch
- u. a. DEC-X-ON-/X-OFF-Protokoll
- Papierbreite 44–397 mm stufenlos
- 3 Schnittstellen: V24, Linienstrom,
- 8 Bit parallel

NEUMÜLLER MESSTECHNIK
Eschenstraße 2 8021 Taufkirchen/München
Telefon 089/61 18-1 Telex 5-22106



An die Superbrain-Besitzer und alle, die es noch werden möchten!

Der Spezialist für Superbrain bietet Ihnen (auch zum Nachrüsten) an:
Neues Betriebssystem CP/M2.2 Vers. 4.0, QD-Version, deutsche Tastatur m. deutschem Zeichensatz, 2 Parallelports, anschließbar an IBM-Rechenanlagen.

Bitte fordern Sie noch heute Unterlagen an.

Intacom
Hohenheimer Straße 11
7022 Leinfelden-Echterdingen 1
Telefon (07 11) 75 10 59

mc quickie – die schnelle Produktanzeige!

mc-quickies sind aktuelle Produktanzeigen, mit denen Firmen ihre Produkte vorstellen. Verantwortlich für den Inhalt sind die Inserenten.

ALTOS

COMPUTER SYSTEMS

OASIS

System Software

softronic

KAISERPASSAGE 2
7500 KARLSRUHE

TELEFON 0721 / 21213
TELEX 782 6633 SOFT - D

Preisgünstiger Matrixdrucker

MDIVP ist ein kleiner leistungsfähiger Matrixdrukker, der auf 60 mm breitem Normalpapier einen ASCII-Zeichensatz von 64 Zeichen ausdruckt. Pro Zeile können 21 Z. in 5 × 7 Matrix mit einer Geschwindigkeit von 50 Zeichen/Sek. ausgedruckt werden. Die Prozessorsteuerung verfügt über eine direkte uP-Schnittstelle und Centronics-Schnittstelle. Preis DM 399.– inkl. MwSt.

F.J. Grimberg Elektronik, Ruthsstr. 24, 6100 Darmstadt, Tel. (0 61 51) 7 68 68 + 71 65 66

GWK GESELLSCHAFT FÜR TECHNISCHE ELEKTRONIK mbH

6809 Multiuser/Multitasking Computersystem

auf Europakarten wurde zur Hannover-Messe vorgestellt

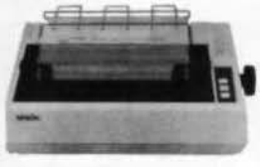
- **CPU-Karte 6809**
- **Floppy-Disk-Controller**
- **Seriell- und Parallel-I/O**
- **AD/DA-Converter**
- **RAM 32K**
- **Eprom Board 16/32 KByte**
- **Bus Board 64/96**

Bitte fordern Sie ausführliche Unterlagen an.

Asternstr. 2, D-5120 Herzogenrath, Telefon (0 24 06) 6 23 94

N E U Epson

MX-80
MX-80 FT



- 40-132 Zeichen/Zeile, 80 Z/Sek.
- Bidirektionaler Druck m. Druckwegoptimierung
- Centronics Parallelschnittstelle Standard; V-24, TTY, IEC u.a. optional
- Groß-, Klein-, Spreitzschrift, Unterlängen, Graphik
- Tabulierung, Formfeed, Fettdruck u.v.a.m. programmierbar
- Geräuscharm (ca. 60 dBA)

Einzelpreis: ab DM 1800.– **inklusive** MwSt. + Zustellung

OEM- und Händlerpreise auf Anfrage

SCHWIND DATENTECHNIK G.M.B.H.

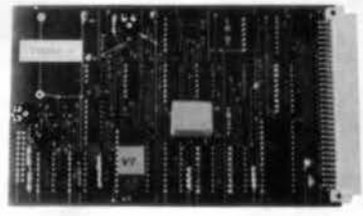
Ebenböckstr. 4, D-8000 München 60
Tel.: (089) 8 34 97 16, Telex: 05-213 097

Messen mit Mikrocomputern

Die preiswerte Alternative, ausbaubar zum kompletten Meßsystem mit jedem Mikrocomputer. Über nur drei Leitungen zu Ihrem Mikrocomputer können Sie bis zu 64 verteilte Meßstellen ohne weiteres Interface auch über weite Entfernungen gezielt abfragen.

Digitalvoltmeter (Europakarte) DM 295.–
mit drei Stellen, LED-Anzeige, Eingangswiderstand > 10 MΩ, Grundmeßbereich –99 mV bis 999 mV, max. Genauigkeit 0,1%, Versorgungsspannung 7–10 V=, mit Meßbusanschluß (s. o.). (Auch im kompletten System mit Personalcomputer ABC80, Epson-Drucker MX80, FKS-Floppy mit 778 kB und Zubehör lieferbar.)

D. Köpke, Rechner zur Kommunikation, Bitzenstraße 11, D-5464 Asbach, Telefon (0 26 83) 45 47



Video 1

- ASCII-**Groß-** und **-Kleinbuchstaben**
- **Austauschbarer Zeichensatz**
- Anschließbar an alle µP-Systeme
- **Invertiertes Bild** mittels Steuerzeichen
- **Blinken** mittels Steuerzeichen
- **Hellschreiben** mittels Steuerzeichen
- **2 Zusatzsignale,** z. B. Farbwiedergabe
- Option: UHF-Modulator auf der Karte
- Sensationeller Preis **DM 268.–** inkl. MwSt.

Ostrowski Mikroelektronik
Zum Römergrund 59, 6501 Wörrstadt
Telefon (0 67 32) 41 78

Neu: ROMBOX FÜR COMMODORE

14 (2 × 7) freie Sockel für ROMs und EPROMs. Umschaltung softwareseitig durch einfachen Poke-Befehl. In einem Programm kann mit allen 14 Sockeln gearbeitet werden!
Anschluß über Flachkabel am Memory Expansion Port. Stromversorgung über den Rechner.
Sofort lieferbar für CBM 3000, 4000 und 8000 für DM 593.– inkl. MwSt.

DATA BECKER GMBH

Merowingerstr. 30 · 4000 Düsseldorf
Tel. 02 11/31 20 85/86 · Telex 08 582 874

SHARP MZ-80K

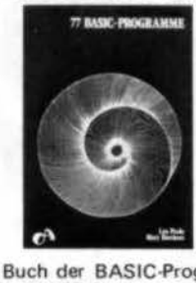
Der supervielseitige Personal-Computer

mit 24 KB-RAM	nur DM	**2098.–**
mit 36 KB-RAM	nur DM	**2160.–**
mit 48 KB-RAM	nur DM	**2220.–**
16 K-RAM zum Selbsteinbau	nur DM	**65.–**
Maschinensprache Buch m. Kass.	DM	**59.–**
Assembler, Debugger, Relocator	DM	**145.–**
PC-1211 BASIC-Taschencomputer	DM	**450.–**
CE-121 Schnittstelle f. PC 1211	DM	**62.–**
CE-122 Drucker m. Cassettenschnittstelle	DM	**298.–**

alfred graumann
Elektronik-Vertrieb
(0 40) 2 50 30 50 · Grevenweg 63 · 2000 Hamburg 26

---

77 BASIC–PROGRAMME



Das erste Buch der BASIC-Programmierserie beschreibt 77 Kurzprogramme, die finanztechnische, mathematische, statistische und verschiedene allgemeine Aufgaben mit zahlreichen Programmierbeispielen und möglichen Anwendungen behandeln.

Verfasser Lon Poole und Mary Borchers, Herausgeber A.Osborne/McGraw-Hill,Inc., in deutscher Sprache, 208 Seiten, Softcover, Preis DM 39,–

te-wi Verlag GmbH, Waldfriedhofstraße30, 8000 München 70,Telefon: (089)37 62 39

mc-minimarkt & programmbörse

Biete an Hardware

Gebr. EDV-Anl. IBM 1130 kompl. mit Leser, Stanzer, Drucker, VB 7000.–; EDV-Anl. Philips P 354 für Magnetkonten mit Zub. VB 1500,–; Magnetband-Einheit Control-Data CD 601 ab 950.–; Ampex Magnetbandlöschgerät 150.–; 8-Kanallocher ADDO 250.–; 8-K-Schnelllocher Facit PE 1500 980.–; div. XY-Schreiber ab 900.–; Siemens Fernschreiber m. Lochstr. ab 250.–; Lochkartensortiergerät Maul Selektor AS-12 490.–; Andimat 3 Meßwerterf. Anl. W&G VB 2000.-; DYMEC Meßwerterf. Anl. VB 1500.–; Olivetti Te 318/Tc 349 790.– dazu V 24 Interface 100.– electronic circuits GmbH, Tel. 0 64 06/40 63

QUME+DIABLO Original Typenräder+Farbbandkassetten. 3M-Disketten, H. Saak, Postf. 25 04 61, 5000 Köln 1, Tel. 31 91 30

PLOTTER APPLE II selbstdim, Fkt-inp, errkorr, Histo. 8K, 9 Befehle PLOTTER 2:I3K, 15 Bef, Pdl Dehn, Kursor, DM 50.– bzw. 90.– + Porto ausf. Beschr. verk: SPK LANDSHUT 630632 Bauer, Klötzlmüllerstr. 176 c, 83 Landshut

Sharp MZ 80 K-Zubehör: Kontrastscheibe aus Acrylglas grün od. orange 20.–; Interface mit Modulator für Fernseher 89.–; Disassembler 59.–; Data-Recorder mit Zählwerk (auch für PET lieferbar) 219.– gegen Vorausscheck od. zzgl. NN bei ec-GmbH, Bahnhofstr. 19, 6301 Wettenberg-Wissmar

16-K-RAM-Satz (1. Wahl, 200ns) für MZ80K, TRS-80, CBM, APPLE: DM 75.– Einbauanl. kostenl. Tel. (05 31) 34 32 98

Verkaufe IBM-Kugelkopf Mod. M25, überholt, mit Interface für TRS 80 für 1600 DM. Tel. (0 22 92) 57 50

Datensichtgerät H1000, 80 Char./24 Z., + Drucker, V24, für alle MC. Zus. 1200 DM VB. Tel. (0 61 81) 6 97 58)

APPLE II: Über 100 Programme, z. B.: Sargon II: 89.90 DM. Gratiskatalog anfordern! PANDASOFT, Kamminer Str. 34, 1000 Berlin 10

Notverk.: cbm-3016 mit Zubehör DM 1800.–. Tel. (0 22 38) 1 42 85

NEUHEIT! Mini-Printer, 40 Z./Zeile, 256 Zeichen inkl. Grafik, univers. anschließbar, ab DM 555.–; GP 80 ab DM 1118.–. Grimberg, Ruthsstr. 24, 6100 Darmstadt, Tel. (0 61 51) 71 65 66

XY-Schreiber, auch def. Tel. (0 91 91) 25 84, Wilk

Basic-Compiler für CBM 3032+3040. Info gegen Freiumschlag. G. Freitag, König-Heinrich-Str. 24, 6930 Eberbach

Videointerface für alle Mikrocomputer. ASCII Groß- und Kleinbuchstaben. Auswechselbarer Zeichensatz. Norm. und invert. Bild. Blinken und 2 versch. Helligkeitsstufen durch Softwaresteuerung. Parallele oder serielle Schnittstelle. Option: UHF-Modulator. Preis ab 268.– inkl. MwSt. Ostrowski, Römergrund 59, 6501 Wörrstadt, Tel. (0 67 32) 41 78, ab 14.00 Uhr

Biete an Software

PASCAL für cbm 30xx Vers. 2.0 jetzt verfügbar, DM 998.–. Erzeugt 6502-Code, kein Interpreter. Ersetzt unsere bislang angebotene Vers. 1.0 und benötigt kein Sprachmodul mehr. Bugs wurden beseitigt, RECORDs implement. REALs ca. Mai kostenfreie Nachlieferung. Version für BASIC 4.0 i. V. phs, Teichstr. 9, 3000 Han. 91

SPEEDY-TAPE und SPEEDY-DATE für PET/cbm. Programme + Daten mit 2500 statt 300 Baud auf dem eingebauten Recorder ohne Hardwareveränderungen! Lädt 8k BASIC in 25 Sek.! Bitte Rechnertyp und RAM-Bereich angeben. Neuer Preis: DM 112.– + MwSt. für beide Programme. phs, Teichstr. 9, 3000 Hannover 91

Tausche

Gebe Programme für CBM 3032/2001 ab und suche Programme für CBM 8032 aller Art, auch Tausch. Programmliste gegen Freiumschlag von M. Zeller, Postfach, 8901 Batzenhofen

Programmtausch Pet und Cbm. Tausche Progr. f. Pet & Cbm. Zur Zeit über 350 Progr. Liste von Jürgen Rued, Eisenbahnstr. 66, 7800 Freiburg, Tel. (07 61) 2 20 12

Verschiedenes

Neu für TRS-80: JOYSTICK-INTERFACE, 2 Kanäle + Treiber + Demo + Manual: DM 69.–, Soft dazu: Flugsimul., Tennis je DM 29.–. Tel. (05 31) 34 32 98, auch abends

Auftrag für Gelegenheitsanzeige in mc

mc-programmbörse

Haben Sie ein Programm entwickelt, das vielleicht auch für andere Mikrocomputer-Anwender interessant sein könnte? Dann können Sie es zu einem günstigen Preis in der **mc-programmbörse** anbieten. Ihr Angebot sollte den Programmnamen, den Verwendungszweck, die Sprache, den Computertyp/KByte, das Medium (Floppy, Kassette, Listing usw.), den Preis und Ihre Anschrift enthalten. Zum Beispiel: „HAUSH: Einkaufshilfe für Lebensmittel durch lfd. Verbrauchskontrolle und Bestandsprüfung in BASIC für AIM-65 oder PC-100, 4 KByte, Kassette, 20.– DM"

mc-minimarkt

Im **mc-minimarkt** veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Preise für Gelegenheitsanzeigen in der mc-programmbörse und im mc-minimarkt:

Private Gelegenheitsanzeige je Druckzeile 5.– DM inkl. MwSt. Chiffregebühr je Anzeige 5.20 DM.

Gewerbliche Gelegenheitsanzeige je Druckzeile 7.– DM inkl. MwSt. (dürfen nicht unter Chiffre erscheinen).

An den Franzis-Verlag, Anzeigenabteilung, Karlstraße 41, 8000 München 2

Name und Adresse _____

_____ Abo.-Nr. _____

_____ _____

Unterschrift Ich zahle sofort nach Rechnungserhalt. Datum

Ich wünsche folgenden Text zu veröffentlichen:

in Spalte: ○ mc-programmbörse suche ☐ Software biete an ☐ Software

in Spalte: ○ mc-minimarkt suche ☐ Hardware biete an ☐ Hardware

Zutreffendes ankreuzen ☐ Tausch ☐ Versch. ☐ Kontakte ☐ Chiffre

Herausgeber: Franzis-Verlag GmbH, Karlstr. 37, 8000 München 2. Postanschrift: Postfach 37 01 20, 8000 München 37. Telefon (0 89) 51 17-1, Telex 5 22 301, Postscheckkonto München 57 58-807.
Verlagsleiter: Peter G. E. Mayer.
Gesellschafter: G. Franz'sche Buchdruckerei G. Emil Mayer KG, München. Persönlich haftender Gesellschafter: Peter G. E. Mayer, Verleger (50 %). Kommanditisten: Ilse Bergmann, Prokuristin (12,5 %); Michael A. Mayer, Betriebswirt (37,5 %); alle wohnhaft in München. Auslandsgesellschaft: Franzis Publishing Co., Michael A. Mayer, 2816 Clay St., San Francisco, CA 94115, USA, Tel. (001 415) 346-2032, Telex 002 559 103 722 019.

Redaktion: Dipl.-Ing. (FH) Herwig Feichtinger (Chefredakteur), Dipl.-Ing. (FH) Rudolf Hofer (fl), Dipl.-Math. Ulrich Rohde. Franzis-Labor: Dipl.-Ing. (FH) Hans Neumayr. Herstellung: Günter Ropertz. Nachdruckrechte, Sonderdrucke, Lizenzen: Siegfried Pruskil.

Anzeigen: Anzeigenleiter: Gerhard Walde. Anzeigen-Verkaufsleiter: Johann Bylek. Disposition: Regina Reinhard, Tel. 0 89/51 17-3 40. Anzeigenpreise nach Preisliste Nr. 1. Anzeigen-Auslandsvertretungen: *USA:* International Media Marketing, 16704 Marquardt Ave., P.O.Box 1234, Cerritos, CA 90701, phone (2 13) 9 26 95 44, telex 9 105 831 412. *Frankreich:* Agence Gustav Elm, 41, Avenue Montaigne, 75008 Paris, phone 01-7 23 32 67. *United Kingdom:* Publicitas Ltd., 525/527 Fulham Road, London SW6 1HF, phone 01-3 85 77 23, telex 9 19 223 publon. *Schweiz:* Exportwerbung AG, Neptunstraße 20, CH-8032 Zürich, Tel. 01-47 46 90, Telex 53 327. *Japan:* International Media Rep. Ltd., 2-29, Toranomon 1-chome, Minatoku, Tokyo 105, phone 5 02-06 56, telex 22 633. *Italien:* Ramcati advertising, Milano San Felice Torre 5, I-20090 Segrate, phone 0 92-7 53 14 45, telex 3 11 010.

Bezug: Vertriebsleiter: Peter Habersetzer. Bestellungen über jede Buchhandlung, die Deutsche Bundespost oder direkt an den Verlag. Abonnement-Kündigung 8 Wochen zum Kalenderjahresende.

Preise:	Einzelheft	Abonnement 1981 (4 Hefte, inkl. Versand)
Deutschland (inkl. 6,5 % MwSt.)	6 DM	20 DM
Österreich	50 öS	186 öS
Schweiz	6,80 sfr	19,50 sfr
USA	–	11.60 $
Sonstiges Ausland	6,50 DM	22 DM

Auslands-Bezug über: *Dänemark:* Jul. Gjellerups Boghandel, Kopenhagen K., Solvgade 87. *Niederlande:* De Muiderkring N. V., Bussum, Nijverheidswerf 17-19-21. *Österreich:* Fachbuch Center Erb, 1061 Wien, Amerlingstr. 1. *Schweiz:* Verlag Thali AG, Hitzkirch, Luzern. *Frankreich:* Librairie Parisienne de la Radio, 43, Rue de Dunkerque, F-75010 Paris.

Verantwortlich für den Textteil: Herwig Feichtinger; für den Anzeigenteil: Gerhard Walde.

Druck: Franzis-Druck GmbH, Karlstraße 35, 8000 München 2, Tel. 0 89/51 17-1.
Imprimé en Allemagne. Printed in Germany.
ISSN 0720-4442.
© 1981 Franzis-Verlag, München.

Die in mc veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, insbesondere das der Übersetzung in fremde Sprachen, vorbehalten. Kein Teil dieser Zeitschrift darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form – durch Fotokopie, Mikrofilm oder andere Verfahren – reproduziert oder in eine von Maschinen, insbesondere Datenverarbeitungsanlagen, verwendbare Sprache übertragen werden. Auch die Rechte der Wiedergabe durch Vortrag, Funk- oder Fernsehsendung, im Magnettonverfahren oder ähnlichem Wege bleiben vorbehalten. Fotokopien für den persönlichen oder sonstigen eigenen Gebrauch dürfen nur von einzelnen Beiträgen oder Teilen daraus als Einzelkopien hergestellt werden. Jede im Bereich eines gewerblichen Unternehmens hergestellte oder benützte Kopie dient gewerblichen Zwecken gem. § 54 (2) UrhG und verpflichtet zur Gebührenzahlung an die VG Wort, Abtlg. Wissenschaft, Goethestr. 49, 8000 München 2, von der die einzelnen Zahlungsmodalitäten zu erfragen sind.

Inserentenverzeichnis

aaa electronic 77
adcomp 17

Compact 81
Computer + Components 67
Computershop 17, 83

Data Becker 17, 78, 79
DTM Datentechnik 78

EGIS 73
Eltec 7

Franzis-Verlag ... 18, 59, 78, Einhefter
Fritz 77
Füssner Computersysteme 16

Gould Instruments Systems 4
Graf 67
Graumann 79
Grimberg 79
GWK 78, 79

HEATH 15
HEW-Computer 31
Hofacker 13
Hopp 78
Hülsewig 73

INSA 73
Intacom 78

Kanis 77
Köpke 79

Ley 73

micro Computer-Zentrum 69
Micro Mag 73
Micropoint 77
Münzenloher 69

NEC 11
Neumüller 78

Ostrowski 78, 79

P + M 67

Rauch 77
Rohde & Schwarz 16, 41
r + r electronic 69

Salhöfer 73
Sinclair 9
SM Softwareverbund 15
Softronic 79
Specovius 14
Spima Computer 77
Syntax 77
Schwind 79

Tektronix 49
Telecom 73
te-wi Verlag 79

Valvo 2
Vobis Data 86

WATANABE 14

Ziegler 70, 71

NEU COMPACT
Die Neuentwicklung aus Berlin für den apple II:
SUPER-PAL-BOARD
● 16 Originalfarben ● 54 Farbvarianten programmierbar ● 16 Grauwerte ● Einzelne Farben über Programm durch Grauwerte ersetzbar ● VHF, Kanal 4, oder über UHF-Modulator
Einführungspreis DM incl. MwSt. **329,–**

NEU COMPACT
Spielen und lernen – lernend spielen mit
ARISTOTELES APPLE
Das Computer-Lehrer- und/oder Quiz-Master-Programm. Für Jeden. Und für jeden Einsatz: Schule, Studium, Kurse, Vereine, Ausstellungen! – Keine Programmierkenntnisse erforderlich. – Speichern der Lerninhalte auf Disk. Für schnellen Zugriff. – 3 Antwortarten: Ausfüllen, Auswählen, Zuordnen – Programm und Beschreibung deutschsprachig
DM incl. MwSt. **450,–**

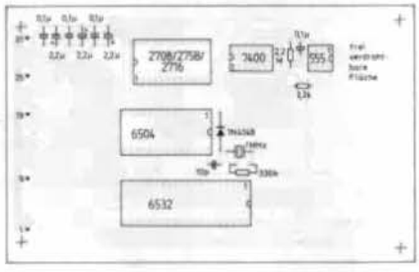
NEU COMPACT
Neuentwicklung aus Berlin für apple II:
Serielles Mehrfunktions-Interface A-V 24
– RS 232 (V 24) -Schnittstelle Handshake (auch Signal Carrier Detect für Modembetrieb) – Umschaltbare Betriebssoftware (4 x 256 Byte). Durch EPROM auch vom Anwender programmierbar. – Zusätzliche Software von 2 KByte im apple Bankselectbereich OC 800 H – OC FFFH auf der Karte in EPROM unterbringbar. – Optoentkoppelte Linienstromschnittstelle vorbereitet. – Sende-/Empfangsleitungen und Handshakeleitung sind auf der Karte per Steckbrücken für die Funktionen Datenendgerät und „Modemersatz" umschaltbar. – Datenübertragungsrate zwischen 55 und 19 200 Baud
incl. MwSt. ab DM **672,35**

COMPACT
In allen Compact-Geschäftsstellen erhältlich oder im Versand über die Hauptverwaltung.
1000 Berlin 15
Fasanenstraße 67 · 030/312 40 91
2000 Hamburg 76
Landwehr 27 · 040/25 80 69
5600 Wuppertal 1
Hofkamp 24 · 0202/44 09 03
2800 Bremen 1
Dobbenweg 8 · 0421/741 76
7500 Karlsruhe 1
Werderstraße 7 · 0721/69 73 73
Hauptverwaltung
1000 Berlin 12
Hardenbergstraße 4/5 · 030/312 40 91
Telex 1 85 592 (treut)

Das lesen Sie in mc 2

Videorecorder als schnelle Datenspeicher

Videorecorder können nicht nur zur Aufzeichnung von Fernsehsendungen, sondern auch zur Datenspeicherung benutzt werden. mc hat für Sie untersucht, ob das zuverlässig funktioniert, welche Geschwindigkeiten und Speicherkapazitäten sich erzielen lassen und wie sich die unterschiedlichen Videokassetten-Systeme dafür bewährt haben. Soviel sei vorweggenommen: Videorecorder scheinen eine interessante Alternative zu Kassettenrecordern und Floppy-Disk-Laufwerken zu sein!



Mädchen für alles

Daß sich viele Aufgaben, für die man früher umfangreiche Logikschaltungen entwickeln mußte, heute besser mit einem Mikrocomputer lösen lassen, dürfte sich inzwischen herumgesprochen haben. Oft genügt dafür sogar schon eine Minimal-Konfiguration – und eine solche stellt mc in Heft 2 vor. Der Europakarten-Computer arbeitet mit der CPU 6504, so daß sich Programme für ihn auf preiswerten Tischcomputern wie PET, CBM, AIM-65 oder PC-100 entwickeln lassen. 16 I/O-Leitungen ermöglichen ihm den Verkehr mit der Außenwelt, und auch ein Interrupt-Timer ist vorhanden.

Z-80-Texteditor

Eine der nützlichsten Anwendungen eines Computers ist die Textverarbeitung. Ein sehr komfortables Programm dafür, das ursprünglich für den TRS-80 geschrieben wurde, sich aber leicht an andere Z-80-Systeme anpassen läßt, finden Sie in mc 2. Besonderer Wert wurde dabei auf einen benutzerfreundlichen Dialogverkehr gelegt. Fehlermeldungen erscheinen in verständlichen deutschen Sätzen. Bei Bedarf ist es leicht möglich, den Befehlssatz individuell zu erweitern.

Außerdem finden Sie in Heft 2...

...die Fortsetzung der Serie „Suchen und Sortieren in Pascal und Basic", die restlichen Adressen des „geknackten TRS-80", Software für Strichcode-Leser sowie viele Programmier- und Hardware-Tips.

Heft 2 erscheint am 29. Juni 1981.

Kontaktkarten und Abonnement-Bestellkarten

Die nebenstehenden mc-Kontaktkarten erleichtern es Ihnen, direkt und schnell Prospekte, Kataloge und Preislisten zu den in Anzeigen angebotenen oder in der Rubrik mc-markt vorgestellten Produkten anzufordern bzw. Bestellungen aufzugeben.

Wenn Sie keinen der Beiträge in mc versäumen wollen, sollten Sie ein Abonnement bestellen. Bitte benützen Sie dazu die Bestellkarte rechts.

Rechnersystem CS-2000

- Preisgünstig
- Modularer Aufbau
- Betriebssystem CP/M 2.x
  (alle Computersprachen verfügbar)
- IEEE 696 (S-100) Bus
- Jede Periphere anschließbar,
  64 K Speicher (erweiterbar für
  Multi-Userbetrieb MP/M)
- 5 und/oder 8 Zoll Floppy (bis zu 4)
-

320 K	1,2 M	2,4 M	4,8 M	10	20 Mbyte
—	—	—	—	—	—
2 x 5"	2 x 8"	2x8" DS	4x8" DS	Harddisk	

- Z 80A CPU 4 MHz mit einer RS 232
  Terminalschnittstelle Standard
- Preis: CS 2000 Rechner mit 8" Doppelfloppy
  incl. 2 Jahren Garantierte Service-Leistung,
  DM 9900,– (ohne Mwst.)
- Das Rechnersystem für Profis und OEMs

Abbildung zeigt:
Terminal TVI 912c
Rechner System CS 2000
auf 8" Doppelfloppylaufwerk

Wir sind in neuen Geschäftsräumen:
COMPUTERSHOP GMBH
Mangoldstraße 10
D-7778 Markdorf
Tel. 0 75 44 / 35 75 Ø
Telex 734 628 msb d

Unser Filialbetrieb:
MSB-COMPUTERLADEN
Unterortstraße 10
D-6236 Eschborn
Tel. 0 61 96 / 4 69 33 Ø

MICRO COMPUTER MSB
MICRO SYSTEM-BERATER

SPEZIELLES UND UNIVERSELLES

COMPUTERGESTEUERTE TYPENRADSCHREIBMASCHINEN UND -DRUCKER ERMÖGLICHEN DIE **SPEZIELLE** TEXT- UND **UNIVERSELLE** DATENVERARBEITUNG AN **EINEM** ARBEITSPLATZ

1. COMMODORE CBM 8026
Speziell für die COMMODORE-Computer.

Komplettpreis: 4.345,- DM

incl. MwSt. ab Lager Aachen anschlußfertig; incl. Interface und Kabel (CBM-IEC) an CBM-Computer.

Traktorführung gegen Aufpreis (s. u.)

Auch als CBM 8027 ohne Tastatur lieferbar.

2. OLYMPIA ESW 100 KSR

Universell für alle Computer mit seriellem Interface.

Komplettpreis:

3.250,- DM

incl. MwSt. ab Lager Aachen, anschlußfertig incl. Interface und Kabel an CBM-Computer

Traktorführung gegen Aufpreis (s. u.)

Auch als ESW 100 RO ohne Tastatur lieferbar.

	MIT TASTATUR		OHNE TASTATUR	
	CBM 8026	OLYMPIA ESW 100 KSR	CBM 8027	OLYMPIA ESW 100 RO
Gehäuse- und Druckmechanik-Hersteller	OLYMPIA	OLYMPIA	OLYMPIA	OLYMPIA
Druckgeschwindigkeit	16 Zeichen/Sek.	16 Zeichen/Sek.	16 Zeichen/Sek.	16 Zeichen/Sek.
Max. Schriftbreite	141 Zeichen	141 Zeichen	141 Zeichen	141 Zeichen
Zusätzliche Traktorführung	398,—	398,—	398,—	398,—
Schnittstelle	CBM-IEC mit zusätzlichen Formatierungen und Anweisungen	seriell (RS 232)	CBM-IEC mit zusätzlichen Formatierungen und Anweisungen	parallel (Centronics)
- für CBM	4.345,—	3.250,—	3.595,—	2.746,—
- für APPLE	— —	3.423,—	— —	2.996,—
- für SORCERER	— —	2.998,—	— —	2.498,—
- für SUPERBRAIN	— —	2.998,—	— —	— —

Preis incl. MwSt. - anschlußfertig incl. Kabel

VOBIS
DATA COMPUTER GMBH

Deutschlands größter Fachversand
für wissenschaftliche Elektronenrechner und Microcomputer

Postfach 1778 - Viktoriastraße 74
5100 Aachen - Telefon 0241/50081 - Telex 0832389